



TUGAS AKHIR - TE 141599

**PEMETAAN DISTRIBUSI GAS POLUTAN MENGGUNAKAN
QUADCOPTER BERBASIS *AUTONOMOUS WAYPOINT*
*NAVIGATION***

IRFAN FACHRUDIN PRIYANTA
NRP 2212 100025

Dosen Pembimbing
Dr. Muhammad Rivai, ST., MT.
Rudy Dikairono, ST., MT.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2016



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT - TE 141599

**MAPPING OF GAS POLLUTANT DISTRIBUTION USING
QUADCOPTER BASED ON AUTONOMOUS WAYPOINT
NAVIGATION**

IRFAN FACHRUDIN PRIYANTA
NRP 2212 100025

Supervisor
Dr. Muhammad Rivai, ST., MT.
Rudy Dikairono, ST., MT.

DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2016

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul “Pemetaan Distribusi Gas Polutan Menggunakan *Quadcopter* Berbasis *Autonomous Waypoint Navigation*” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juni 2016

Irfan Fachrudin Priyanta
2212100025

**PEMETAAN DISTRIBUSI GAS POLUTAN MENGGUNAKAN
QUADCOPTER BERBASIS AUTONOMOUS WAYPOINT
NAVIGATION**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan Untuk
Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Elektronika
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Dosen Pembimbing I,



Dr. Muhammad Rivai, ST., MT.

NIP: 196904261994031003

Dosen Pembimbing II,



Rudy Dikairono, ST., MT.

NIP: 198103252005011002



PEMETAAN DISTRIBUSI GAS POLUTAN MENGUNAKAN QUADCOPTER BERBASIS *AUTONOMOUS WAYPOINT NAVIGATION*

Nama : Irfan Fachrudin Priyanta
Pembimbing I : Dr. Muhammad Rivai, ST., MT.
Pembimbing II : Rudy Dikairono, ST., MT.

ABSTRAK

Pencemaran udara adalah suatu kondisi dimana kualitas udara menjadi rusak dan terkontaminasi oleh zat-zat, baik yang tidak berbahaya maupun berbahaya bagi makhluk hidup. Seiring dengan perkembangan dan pembangunan industri di Indonesia, berdampak pada meningkatnya tingkat pencemaran udara. Sistem monitoring emisi gas polutan industri secara umum dilakukan dengan cara manual menggunakan sensor gas pada titik-titik tertentu. Hal ini memakan waktu dan biaya cukup banyak untuk per unit modul.

Quadcopter merupakan salah satu jenis *Unmanned Aerial Vehicle* (UAV) yang mampu bergerak secara otomatis sesuai dengan sistem *tracking waypoint*. Sistem *tracking waypoint* merupakan sistem navigasi berdasarkan posisi GPS dan kompas, sehingga *quadcopter* dapat berjalan secara auto. Implementasi sensor gas semikonduktor dapat menunjang *quadcopter* untuk mengukur kadar gas di udara. Tujuan tugas akhir ini adalah *quadcopter* dapat diaplikasikan sebagai alat monitoring gas polutan secara otomatis.

Hasil dari pengujian yang dilakukan pada tugas akhir ini adalah *quadcopter* dapat mendeteksi kadar gas CO di udara secara otomatis berdasarkan titik-titik *waypoint*. Data polutan gas dapat dilihat langsung melalui web server *plotly* secara online. Lama waktu terbang *quadcopter* hingga 10 menit menggunakan baterai lipo 4 sel. Error pembacaan sensor gas adalah 9,4% dari alat pengukur CO-meter menggunakan model matematis regresi power. Error GPS saat pengujian bervariasi antara 20-66 cm dari titik *waypoint* yang dideklarasikan. Hasil pemetaan kadar gas ditampilkan dalam *3D analyzer google earth*.

Kata Kunci : Pencemaran Udara, *Quadcopter*, Sensor Gas Semikonduktor, *Waypoint*.

MAPPING OF GAS POLLUTANT DISTRIBUTION USING QUADCOPTER BASED ON AUTONOMOUS WAYPOINT NAVIGATION

Name : Irfan Fachrudin Priyanta
1st Advisor : Dr. Muhammad Rivai, ST., MT.
2nd Advisor : Rudy Dikairono, ST., MT.

ABSTRACT

Air pollution is a condition in which the air quality becomes damaged and contaminated by substances that are either dangerous or harmful to living things. Along with the development and construction industry in Indonesia resulted in increased levels of air pollution. Generally, the monitoring system of industrial pollutant emissions was done by using gas sensors at certain points. This takes time and costs quite a lot to per unit modules.

Quadcopter is one of Unmanned Aerial Vehicle (UAV) which capable to move automatically according to the waypoint tracking system. Waypoint tracking system is a navigation system based on GPS positioning and compass, so the quadcopter can work automatically. Implementation of the semiconductor gas sensor can support quadcopter to measure levels of the gas in the air. The purpose of this final project is to develop quadcopter that can be applied as an automatically monitoring tool for pollutant gases .

The results of tests which have done in this final project is quadcopter can detect levels of CO gas in the air automatically based on the waypoints. Flying time of the quadcopter is up to 10 minutes using lipo 4-cell battery. Error reading of gas sensor is 9.4% from the CO-meter gauges using power regression mathematical model. The GPS error varied between 20-66 cm from the waypoints declared. The result of gas's mapping can be displayed in 3D vizualizer google earth.

Keyword: *Air Pollution, Quadcopter, Semiconductor Gas Sensor, Waypoint.*

KATA PENGANTAR

Puji syukur atas rahmat yang diberikan oleh Tuhan Yang Maha Esa. Karena berkat rahmat dan hidayah-Nya penulis dapat menyelesaikan penelitian ini. Selama pelaksanaan penelitian Tugas Akhir ini, penulis mendapatkan bantuan dari berbagai pihak baik dukungan secara moril maupun materiil. Terima kasih yang sebesar-besarnya penulis sampaikan kepada berbagai pihak yang mendukung dan membantu dalam tugas akhir ini, diantaranya :

1. Kedua orang tua tercinta, Bapak Ir. Dwi Priyanta, M.SE. dan Ibu Koesmiyati, S.E. yang tidak pernah putus untuk seluruh do'a, nasihat, motivasi, dan dukungannya.
2. Dr. Muhammad Rivai, ST., MT. selaku dosen pembimbing pertama, atas bimbingan, inspirasi, pengarahan, dan motivasi yang diberikan selama pengerjaan penelitian tugas akhir ini.
3. Rudy Dikairono, ST., MT. selaku dosen pembimbing kedua, atas bimbingan, inspirasi, pengarahan, dan motivasi yang diberikan selama pengerjaan penelitian tugas akhir ini.
4. Teman-teman asisten Lab B202 yang senantiasa membantu dan memberikan semangat dalam mengerjakan tugas akhir.
5. Teman-teman asisten Lab B402 yang senantiasa membantu dan memberikan semangat dalam mengerjakan tugas akhir.
6. Teman-teman Tim Robot ITS yang senantiasa turut membantu dan memberikan dukungan dalam mengerjakan tugas akhir.

Penulis sadar bahwa Tugas Akhir ini masih belum sempurna dan masih banyak hal yang perlu diperbaiki. Saran, kritik dan masukan baik dari semua pihak sangat membantu penulis terutama untuk berbagai kemungkinan pengembangan lebih lanjut.

Surabaya,

Irfan Fachrudin Priyanta
2212100025

DAFTAR ISI

Halaman

ABSTRAK.....	i
ABSTRACT.....	iii
KATA PENGANTAR	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xiii
BAB I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Batasan Masalah	3
1.5 Metodologi Penelitian	3
1.6 Sistematika Penulisan	5
1.7 Relevansi.....	5
BAB II.TEORI PENUNJANG.....	7
2.1 Pencemaran Udara	7
2.1.1 Indeks Standar Pencemaran Udara	8
2.1.2 Gas Karbon Monoksida (CO).....	9
2.2 Sensor Semikonduktor	9
2.2.1 Sensor Gas MQ-7	11
2.2.2 Rangkaian Dasar Pengukuran Sensor Gas	12
2.3 <i>Quadcopter</i>	12
2.3.1 Remote Control 2.4 GHz	13
2.3.2 Baterai Lythium Pollymer (Lipo)	15
2.3.3 Brushless DC (BLDC) Motor	16
2.3.4 Propeller.....	17
2.3.5 Electronic Speed Controller (ESC).....	17
2.4 Navigasi Pada <i>Quadcopter</i>	18
2.4.1 Global Positioning System (GPS).....	18
2.4.2 Sistem Inertial Measurement Unit (IMU).....	19
2.4.3 PIXHAWK Flight Controller	20
2.4.4 Software Mission Planner.....	21
2.5 STM32F4.....	21
2.6 Raspberry Pi 2.....	22

2.7	Pengendali Propotional Integral Derivatif (PID)	23
BAB III. PERANCANGAN SISTEM.....		25
3.1	Diagram Blok Sistem.....	25
3.2	Perancangan Perangkat Elektrik	27
3.2.1	Power Module Distributor.....	28
3.2.2	Power Supply	28
3.2.3	Rangkaian Sensor Gas MQ-7	28
3.2.4	STM32F4	30
3.2.5	Raspberry Pi 2	31
3.2.6	Pixhawk.....	31
3.3	Perancangan Mekanik <i>Quadcopter</i>	32
3.3.1	Perhitungan Beban <i>Quadcopter</i>	32
3.3.2	Desain Awal <i>Quadcopter</i>	33
3.4	Perangkat lunak	33
3.4.1	Perangkat Lunak STM32F4	34
3.4.1.1	Perangkat Lunak ADC STM32F4	34
3.4.1.2	Perangkat Lunak Remote Control STM32F4	36
3.4.2	Perangkat Lunak <i>Mission Planner</i>	37
BAB IV. PENGUJIAN DAN ANALISIS SISTEM.....		39
4.1	Realisasi Desain <i>Quadcopter</i>	39
4.2	Pengujian Perangkat Keras	40
4.2.1	Pengujian <i>Supply Buck Converter</i>	41
4.2.2	Pengujian Kalibrasi Sensor	41
4.2.2.1	Prosedur Kalibrasi Awal Sensor Gas MQ-7	41
4.2.2.2	Pencarian Model Matematis Kalibrasi	44
4.2.2.3	Perbandingan Nilai Error Regresi Power	48
4.2.2.4	Perbandingan Nilai Sensor Gas MQ-7	51
4.2.3	Pengujian <i>Remote Control</i>	52
4.2.4	Pengujian GPS dan <i>Waypoint Quadcopter</i>	54
4.3	Perangkat Lunak Raspberry Pi 2	55
4.3.1	Program Plotly Raspberry Pi 2	56
4.4	Pengujian Seluruh Sistem	57
4.4.1	Pengujian Sistem Ke-1	57
4.4.2	Pengujian Sistem Ke-2	60
4.4.3	Pengujian Sistem Ke-3	64
4.4.4	Pengujian Sistem Ke-4	69
4.5	Evaluasi Sistem.....	75
BAB V. PENUTUP		77
5.1	Kesimpulan	77

5.2 Saran	77
DAFTAR PUSTAKA	79
LAMPIRAN.....	81
BIODATA PENULIS	87

.....*Halaman ini sengaja dikosongkan*.....

DAFTAR GAMBAR

Gambar 2.1 Ilustrasi Penyerapan O ₂ Oleh Sensor	9
Gambar 2.2 Bentuk Penghalang Potensial Kristal Mikro Sno ₂	10
Gambar 2.3 Sensor Gas MQ-7	11
Gambar 2.4 Grafik Sensitifitas Sensor	11
Gambar 2.5 Rangkaian Pengukuran Dasar Sensor	12
Gambar 2.6 Skema Gerakan Quadcopter.	13
Gambar 2.7 Quadcopter	13
Gambar 2.8 Remote Control 2.4 Ghz merk Futaba	14
Gambar 2.9 Konfigurasi Baterai Lithium Polymer	15
Gambar 2.10 Konfigurasi Putaran Brushless Motor	16
Gambar 2.11 Perbandingan Gaya Dorong <i>Propeller</i>	17
Gambar 2.12 Konfigurasi <i>Electronic Speed Controller</i>	17
Gambar 2.13 Konfigurasi GPS M8N	19
Gambar 2.14 Diagram Blok Sistem GNC	19
Gambar 2.15 Pixhawk Flight Controller	20
Gambar 2.16 Software Mission Planner	21
Gambar 2.17 STM32F4-Discovery	22
Gambar 2.18 Raspberry Pi 2 Model B	23
Gambar 2.19 Implementasi PID	23
Gambar 3.1 Diagram Blok Sistem Quadcopter	26
Gambar 3.2 Komponen Sistem Quadcopter	27
Gambar 3.3 Power Module Pixhawk	28
Gambar 3.4 Rangkaian Power Supply Dan <i>Buck Converter</i>	28
Gambar 3.5 Rangkaian Sensor MQ-7	29
Gambar 3.6 Konfigurasi Kalibrasi Sensor Gas MQ-7	29
Gambar 3.7 Konfigurasi Raspberry Pi 2	31
Gambar 3.8 Konfigurasi Pixhawk	32
Gambar 3.9 Desain <i>Quadcopter</i> pada <i>Auto Cad</i>	33
Gambar 3.10 Blok Sistem Perangkat Lunak <i>Quadcopter</i>	34
Gambar 3.11 Flowchart Program ADC STM32F4	35
Gambar 3.12 Flowchart Program <i>Remote Control</i>	36
Gambar 3.13 Tampilan Awal <i>Software Mission Planner</i>	37
Gambar 4.1 <i>Quadcopter</i> yang Diterapkan	39
Gambar 4.2 Realisasi Desain <i>Quadcopter</i> yang Telah Dirakit	40
Gambar 4.3 Pengujian Vin 12 volt	41
Gambar 4.4 Alat Co-meter Model 7701 <i>Az Instrument</i>	42
Gambar 4.5 Tampilan Awal CO-meter Model 7701	42

Gambar 4.6 Tampilan CO-meter Setelah Kalibrasi	43
Gambar 4.7 Pengujian Kalibrasi Sensor Menggunakan CO-meter	43
Gambar 4.8 Tegangan Output Sensor di Multimeter	44
Gambar 4.9 Hasil Data Tabel 4.2 Dalam Bentuk Grafik	46
Gambar 4.10 Hasil Kalibrasi Kurva Regresi Linier	47
Gambar 4.11 Fungsi Algoritma dari Kalibrasi Sensor Gas	47
Gambar 4.12 Grafik Fungsi Algoritma dari Sampel Regresi Power	48
Gambar 4.13 Grafik Fungsi Algoritma dari Sampel Regresi Linier	49
Gambar 4.14 Grafik Fungsi Algoritma dari Sampel Regresi Eksponen	50
Gambar 4.15 Grafik Fungsi Algoritma dari Sampel Regresi Log	50
Gambar 4.16 Pengujian Nilai Input Sinyal PWM	52
Gambar 4.17 Perintah <i>Waypoint</i> yang Dideklarasikan	54
Gambar 4.18 Program Pembacaan Data Serial USB	55
Gambar 4.19 Program <i>Plotly</i> Raspberry Pi 2	56
Gambar 4.20 Hasil Pembacaan Kadar Gas di <i>Plotly</i>	57
Gambar 4.21 <i>Quadcopter</i> Saat Melakukan Awal Misi Pertama	58
Gambar 4.22 Lokasi Pengujian Sistem Pertama	58
Gambar 4.23 Hasil Data Logger Pengujian Ketiga di <i>Plotly</i>	59
Gambar 4.24 Lokasi Pengujian Sistem Kedua	60
Gambar 4.25 Deklarasi <i>Waypoint</i> Kedua	61
Gambar 4.26 Hasil Data Logger Pengujian Kedua di <i>Plotly</i>	63
Gambar 4.27 Hasil 3D <i>Analyzer Google Earth</i>	63
Gambar 4.28 Lokasi Pengujian Sistem Ketiga	64
Gambar 4.29 Deklarasi <i>Waypoint</i> Ketiga	65
Gambar 4.30 Error Ketinggian <i>Quadcopter</i>	66
Gambar 4.31 Hasil Data Logger Pengujian Ketiga di <i>Plotly</i>	67
Gambar 4.32 Hasil 3D <i>Analyzer Google Earth</i>	68
Gambar 4.33 Lokasi Pengujian Sistem Ketiga	69
Gambar 4.34 Hasil Pembacaan Co-meter di Polutan	69
Gambar 4.35 Deklarasi <i>Waypoint</i> Ketiga	70
Gambar 4.36 Error Ketinggian <i>Quadcopter</i>	71
Gambar 4.37 Hasil Data Logger Pengujian Keempat di <i>Plotly</i>	72
Gambar 4.38 Pengujian <i>Quadcopter</i> saat di <i>Ground</i>	74
Gambar 4.39 Hasil 3D <i>Analyzer Google Earth</i>	75

DAFTAR TABEL

Tabel 1.1 Beban Pencemar Udara dari Sumber Tidak Bergerak.....	1
Tabel 2.1 Komposisi Udara Bersih	7
Tabel 2.2 Udara Bersih dan Udara Kotor Menurut WHO.....	8
Tabel 2.3 Batas Indeks Standar Pencemaran Udara	8
Tabel 3.1 Estimasi Berat <i>Quadcopter</i>	32
Tabel 4.1 Perbedaan Tegangan Multimeter dengan Tegangan ADC	44
Tabel 4.2 Hasil Kalibrasi Sensor Gas MQ-7	45
Tabel 4.3 Sampel Data yang Digunakan Perhitungan Error	48
Tabel 4.4 Perhitungan Error Regresi Power.....	49
Tabel 4.5 Perhitungan Error Regresi Linier	49
Tabel 4.6 Perhitungan Error Regresi Eksponensial.....	50
Tabel 4.7 Perhitungan Error Regresi Logaritmik	51
Tabel 4.8 Perbandingan Nilai Sensor Gas MQ-7	52
Tabel 4.9 Nilai Sinyal PWM Masing-Masing Channel.....	53
Tabel 4.10 Nilai GPS Pengujian <i>Waypoint</i>	55
Tabel 4.11 Data Kadar Gas PPM Saat Pengujian Pertama	59
Tabel 4.12 Nilai GPS Pengujian <i>Waypoint</i> Kedua.....	61
Tabel 4.13 Data Kadar Gas PPM Saat Pengujian Kedua	62
Tabel 4.14 Nilai GPS Pengujian <i>Waypoint</i> Ketiga.....	65
Tabel 4.15 Nilai Sensor Barometer saat Pengujian Ketiga	66
Tabel 4.16 Data Kadar Gas PPM Saat Pengujian Ketiga.....	67
Tabel 4.17 Nilai GPS Pengujian <i>Waypoint</i> Keempat.....	70
Tabel 4.18 Nilai Sensor Barometer saat Pengujian Ketiga	71
Tabel 4.19 Data Kadar Gas PPM Saat Pengujian Keempat	72
Tabel 4.20 Data Kadar Gas PPM Saat Pengujian di <i>Ground</i>	74

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pencemaran udara menurut Peraturan Pemerintah Republik Indonesia No. 41 Tahun 1999 mengenai Pengendalian Pencemaran Udara, adalah masuknya atau dimasukkannya zat, energi, dan/atau komponen lain ke dalam udara ambien oleh kegiatan manusia sehingga mutu udara ambien turun sampai ke tingkat tertentu yang menyebabkan udara ambien tidak memenuhi fungsinya. Kota Surabaya sebagai kota metropolitan di Indonesia menyumbang berbagai macam pencemaran udara. Pencemaran udara di Kota Surabaya, 70% diakibatkan oleh transportasi, sisanya disebabkan oleh industri dan limbah (persampahan).

Sektor industri merupakan penyumbang pencemaran terutama yang berhubungan dengan proses kegiatan industri tersebut. Emisi industri turut menyumbang terhadap penurunan kualitas udara karena belum semua industri memiliki alat pengendali pencemar udara yang memadai. Industri-industri besar yang menggunakan bahan bakar fosil banyak menghasilkan gas buang yang dapat menyebabkan pencemaran udara [1].

Tabel 1.1 Beban Pencemar Udara dari Sumber Tidak Bergerak (Industri Pengolahan)

SUMBER KEGIATAN	KONSENTRASI (TON/TAHUN)					
	Debu	SO ₂	NO _x	HC	CO	Lainnya
Makanan	4038.19	0	0	0	0	0,013
Minuman	0.016	0	0	0	0	0
Kayu	38.61	0	18.02	0	0	0
Kimia Dasar	26.9	0	0	215.2	0	0
Mineral Non Logam	1	0	0	0	0	0
Logam Dasar	28.8	0	0	0	232	0
Hasil Olahan Logam	0	0	0	0	0	0
Tekstil	598.6	0	0	0	0	0
Total	4732.116	0	18.02	215.2	232	0,013

Sumber : Dinas Perindustrian & Penanaman Modal
Diolah oleh BPLH, 2008

Aktivitas industri di Surabaya yang berkembang di sekitar Rungkut, Wonokromo, dan Gresik telah menunjukkan penurunan kualitas udara.

Setiap pabrik selayaknya mengikuti Keputusan MENLH Nomor KEP-13/MENLH/03/1995 tentang Baku Mutu Emisi Sumber Tidak Bergerak. Dimana setiap pabrik wajib memenuhi baku mutu emisi udara menggunakan alat atau teknologi terbaru. Sesuai dengan Keputusan MENLH tersebut, setiap industri wajib memiliki alat pengendali pencemar udara yang memadai. Beberapa pabrik telah memasang sensor gas sebagai alat *monitoring* pencemaran udara, namun sebagian besar masih melanggar aturan. Sehingga pemkot melakukan inspeksi ke setiap pabrik yang memakan biaya cukup besar dan waktu yang cukup lama.

Berdasarkan tabel 1 menunjukkan bahwa senyawa SO_x, NO_x, CO, HC, dan debu merupakan kontributor penting dalam pencemaran udara. Teknologi sensor gas seperti *electronic nose*, yang mampu mendeteksi dan memilah jenis senyawa di udara. Sensor tersebut memiliki beberapa jenis seperti sensor gas MQ, TGS2600, TGS2610, TGS2210, Figaro, dan TGS4161 memiliki kehandalan untuk mendeteksi kandungan gas-gas tertentu seperti H₂S, CO, NO_x, HC, CH₄, dan SO_x.

Dampak negatif akibat pencemaran udara cukup berat terhadap lingkungan terutama kesehatan manusia yaitu dengan menurunnya fungsi paru, peningkatan penyakit pernapasan, dampak karsinogen dan beberapa penyakit lainnya. Selain itu pencemaran udara dapat menimbulkan bau, kerusakan materi, gangguan penglihatan dan dapat menimbulkan hujan asam yang merusak lingkungan.

Unmanned Aerial Vehicle (UAV) merupakan jenis pesawat tanpa awak yang berguna untuk melakukan sistem *monitoring* pada suatu wilayah. Vertical Take Off Landing (VTOL) merupakan salah satu jenis UAV yang menggunakan banyak baling-baling sebagai penggerakannya. Salah satu jenis VTOL adalah *Quadcopter* dimana kestabilan dalam melakukan pemantauan jarak jauh dan pengambilan data menjadi nilai positif UAV jenis VTOL. *Quadcopter* memiliki fitur *Tracking Waypoint* dimana *Quadcopter* mampu bergerak secara otomatis mengikuti panduan *Waypoint* GPS (*Global Positioning System*) yang dideklarasikan oleh user [2].

Penelitian tugas akhir ini dirancang untuk mengaplikasikan *Quadcopter*, dengan menggunakan STM32F4 untuk proses pembacaan sensor gas TGS dan data GPS quadcopter. Maka terbentuklah *quadcopter* untuk *monitoring* kadar polutan gas di udara. Sehingga diharapkan mampu menjadi solusi untuk pemetaan distribusi polutan gas pada kawasan industri di Kota Surabaya secara efisien dan mudah.

1.2 Perumusan Masalah

Permasalahan pada tugas akhir ini adalah sebagai berikut :

1. Bagaimana mendeteksi kadar polutan gas CO di kawasan industri
2. Bagaimana memonitor kadar polutan gas di kawasan industri secara otomatis
3. Bagaimana *quadcopter* menghasilkan data berupa kadar polutan gas sesuai dengan titik *waypoint* yang dituju

1.3 Tujuan Penelitian

Penelitian pada tugas akhir ini bertujuan sebagai berikut :

1. Implementasi sensor gas jenis MQ sebagai sensor untuk mendeteksi kadar polutan gas di kawasan industri.
2. Perancangan *quadcopter* menggunakan metode *Autonomous Waypoint Navigation* untuk melakukan *monitoring* kadar polutan gas secara otomatis.
3. *Quadcopter* mampu memetakan distribusi polutan gas di udara pada ketinggian tertentu dan menghasilkan data kadar dan lokasi polutan gas yang disimpan pada *data logger*.

1.4 Batasan Masalah

Batasan masalah dari tugas akhir ini adalah sebagai berikut:

1. Gas yang dideteksi sebagai indikator polutan adalah gas karbon monoksida (CO).
2. Sensor gas yang digunakan berupa sensor gas semikonduktor jenis MQ-7.
3. Proses pengolahan data sensor dan *Tracking Waypoint* dari *Quadcopter* menggunakan Kontroller *Pixhawk* dan STM32F4 *Discovery*.
4. Pengambilan data dilakukan pada sebuah lapangan dengan polutan asap hasil pembakaran.

1.5 Metodologi Penelitian

Dalam penyelesaian tugas akhir ini digunakan metodologi sebagai berikut:

1. Studi literatur

Pada tahap studi literatur dilakukan pengumpulan dasar teori yang menunjang dalam penulisan tugas akhir. Dasar teori tersebut diambil

dari artikel-artikel di internet, forum-forum diskusi, dan *paper* internasional.

2. Perancangan *Hardware* Elektronik

Setelah mempelajari literatur yang ada, selanjutnya merupakan tahap merancang sebuah sistem elektronika dari quadcopter, sensor TGS 2600/2201, dan sensor gas MQ. Sehingga menjadi sebuah kesatuan sistem elektronika yang terintegrasi.

3. Perancangan Mekanik *Quadcopter*

Pada tahap ini, mekanik *quadcopter* dirancang agar mampu mengangkut beban sensor-sensor gas dan kamera. Pemilihan bahan dalam pembuatan *frame quadcopter* menjadi faktor penting penentu kinerja *quadcopter*. Spesifikasi dari *Electronic speed controller*, *brushless motor*, *propeler*, dan baterai Lipo menjadi indikator penting agar *quadcopter* mampu melaju secara efisien dalam mode otomatis.

4. Perancangan *Software*

Pada tahap perancangan *software* dibahas mengenai algoritma pemrograman untuk pengolahan data GPS dari *Pixhawk Flight Controller*, sensor gas MQ-7 yang disimpan pada *data logger*. Selain itu perancangan sistem *database* di komputer. Setelah didapatkan semua data pada *data logger* selanjutnya adalah menyajikan data dalam sebuah model plot pemetaan.

5. Perancangan Sistem

Setelah melakukan riset dari referensi yang berkaitan dengan penerjaan tugas akhir ini, langkah berikutnya adalah melaksanakan perancangan sistem yang akan digunakan dalam implementasi *hardware*. Pada tahap ini penggabungan antara *hardware* elektronik dan *software* pada sebuah *quadcopter* yang telah dirancang.

6. Pengujian Sistem

Proses pengujian sistem dilakukan dengan menghubungkan semua *hardware* yang dibutuhkan untuk terbentuknya sistem *quadcopter* untuk mendeteksi kadar polutan di udara. Cara kerja dari sistem quadcopter ini adalah ketika quadcopter dinyalakan maka quadcopter dikalibrasi terlebih dahulu menggunakan *remote control* untuk memilih mode manual atau

otomatis. Ketika *pixhawk* menerima perintah dengan kode tertentu dari *remote control* maka data diteruskan pada STM32F4 untuk mengubah mode. Apabila mode otomatis, maka sensor-sensor gas akan mulai memindai kadar polutan gas di udara dan GPS akan mulai memindai posisi pembacaan gas.

7. Penulisan Laporan Tugas Akhir

Tahap penulisan laporan tugas akhir dilakukan pada saat tahap pengujian sistem dimulai serta setelahnya.

1.6 Sistematika Penulisan

Laporan tugas akhir ini terdiri dari Lima Bab dengan sistematika penulisan sebagai berikut:

- Bab 1 : Pendahuluan

Bab ini meliputi latar belakang, perumusan masalah, tujuan penelitian, batasan masalah, metodologi, sistematika penulisan, dan relevansi.

- Bab 2 : Dasar Teori

Bab ini menjelaskan tentang dasar-dasar teori yang dibutuhkan dalam pengerjaan tugas akhir ini, yang meliputi teori dasar sensor semikonduktor, kendali *PID*, STM32F4, dan *servo controller*.

- Bab 3: Perancangan Sistem

Bab ini menjelaskan tentang perencanaan sistem perangkat keras elektrik dan mekanik, serta perangkat lunak. Bab ini juga berisi menjelaskan tentang prosedur pengujian yang dilakukan dalam penelitian.

- Bab 4 : Pengujian dan Pembahasan Sistem

Bab ini menjelaskan tentang hasil yang didapat dari pengujian tiap Blok sistem secara keseluruhan

- Bab 5 : Penutup

Bab ini menjelaskan tentang kesimpulan meliputi kekurangan-kekurangan pada kerja alat dari hasil analisa serta saran untuk pengembangan ke depan.

1.7 Relevansi

Hasil dari tugas akhir ini diharapkan dapat memberikan manfaat sebagai berikut:

1. Dapat digunakan sebagai alat pendeteksi kadar polutan gas CO pada kawasan industri maupun komersial.

2. Sebagai dasar penelitian lebih lanjut, agar dapat lebih dikembangkan.

BAB II

TEORI PENUNJANG

2.1 Pencemaran Udara

Udara merupakan salah satu unsur alam yang pokok bagi makhluk hidup yang ada di muka bumi terutama manusia. Tanpa udara yang bersih maka manusia akan terganggu terutama kesehatannya yang pada akhirnya dapat menyebabkan kematian. Komposisi udara bersih menurut *Environmental Chemistry, Air, and Water Pollution* sebagai berikut :

Tabel 2.1 Komposisi udara bersih

Jenis gas	Formula	Konsentrasi (% volume)	Ppm
1. Nitrogen	N ₂	78,08	780,800
2. Oksigen	O ₂	20,95	209,500
3. Argon	Ar	0,934	9,340
4. Carbon Dioksida	CO ₂	0,0314	314
5. Neon	Ne	0,00812	18
6. Helium	He	0,000524	5
7. Methana	CH ₄	0,0002	2
8. Krypton	Kr	0,000114	1

Sumber : *Environmental Chemistry, Air and Water Pollution*

Menurut Peraturan Pemerintahan RI No. 41 Tahun 1999, Pencemaran udara adalah masuknya atau dimasukkannya zat, energi, dan/atau komponen lain ke dalam udara ambien oleh kegiatan manusia, sehingga mutu udara ambien turun sampai ke tingkat tertentu yang menyebabkan udara ambien tidak dapat memenuhi fungsinya.

Kualitas udara ambien dari suatu daerah ditentukan oleh daya dukung alam daerah tersebut serta jumlah sumber pencemaran atau beban pencemaran dari sumber yang ada di daerah tersebut. Zat-zat yang dikeluarkan oleh sumber pencemar ke udara dan dapat mempengaruhi kualitas udara antara lain gas Karbon Monoksida (CO), Nitrogen Oksida (NO_x), Sulfur Dioksida (SO₂), debu serta kandungan Timah Hitam (Pb) dalam debu. Suatu daerah dikatakan tercemar apabila memiliki kandungan udara sebagai berikut :

Tabel 2.2 Udara bersih dan udara kotor menurut WHO

Parameter	Udara Bersih	Udara Tercemar
Bahan Partikel	0,01 – 0,02 mg/m ³	0,07 – 0,7 mg/m ³
SO ₂	0,003 – 0,02 ppm	0,02 – 2 ppm
CO	< 1 ppm	5 – 200 ppm
NO ₂	0,003 – 0,02 ppm	0,02 – 0,1 ppm
CO ₂	310 – 330 ppm	350 – 0,1 ppm
Hidrokarbon	< 1 ppm	1 – 20 ppm

2.1.1 Indeks Standar Pencemaran Udara

Indeks Standar Pencemar Udara (ISPU) adalah angka yang tidak mempunyai satuan yang menggambarkan kondisi mutu udara ambien di lokasi tertentu, yang didasarkan kepada dampak terhadap kesehatan manusia, nilai estetika dan makhluk hidup lainnya (Peraturan Pemerintahan RI No. 41 Tahun 1999).

Tabel 2.3 Batas indeks standar pencemaran udara (Keputusan Kepala Bapedal No. 107 Tahun 1997)

ISPU	PMI0	SO ₂	CO	O ₃	NO ₂
0	0	0	0	0	0
50	50	80	5	120	282
100	150	365	10	235	565
200	350	800	17	400	1130
300	420	1600	34	800	2260
400	500	2100	46	1000	3000
500	600	2620	57,5	1200	3750

Cara penghitungan hasil pengukuran udara ambien yang dikonversikan dalam indeks standar pencemar udara adalah sebagai berikut :

$$I = \frac{Ia - Ib}{Xa - Xb} (Xx - Xb) + Ib \quad (2.1)$$

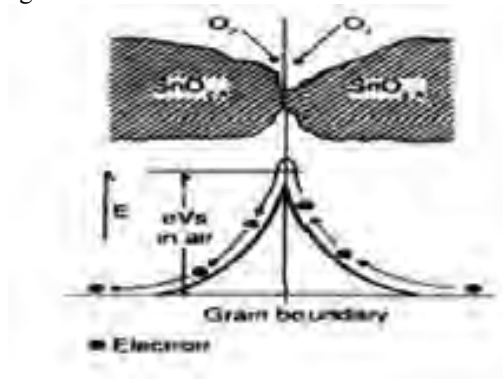
I = ISPU terhitung, Ia = ISPU Batas Atas, Ib = ISPU Batas Bawah, Xa = Ambien Batas Atas, Xb = Ambien Batas Bawah, Xx = Kadar Ambien nyata hasil pengukuran

2.1.2 Gas Karbon Monoksida (CO)

Karbon dan Oksigen dapat bergabung membentuk senyawa karbon monoksida (CO) sebagai hasil pembakaran yang tidak sempurna dan karbon dioksida (CO_2) pembakaran sempurna. Karbon monoksida merupakan senyawa yang tidak berbau, tidak berasa dan pada suhu udara normal berbentuk gas yang tidak berwarna. Karakteristik biologik yang paling penting dari CO adalah kemampuannya untuk berikatan dengan haemoglobin, pigmen sel darah merah yang mengangkut oksigen ke seluruh tubuh. Sifat ini menghasilkan pembentukan karboksihaemoglobin (HbCO) yang 200 kali lebih stabil dibandingkan oksihemoglobin (HbO_2).

2.2 Sensor Semikonduktor

Sensor gas adalah sensor yang berfungsi untuk mengukur senyawa gas polutan yang ada di udara seperti *karbon monoksida*, *hidrokarbon*, *nitrooksida*, dan lain-lain. Sensor gas semikonduktor memiliki banyak jenis, tentunya dibedakan oleh sensitivitas sensor tersebut. Berdasarkan gambar 2.1 prinsip kerja dari sensor ini semakin tinggi konsentrasi gas maka resistansinya semakin rendah. Sensor gas semikonduktor terbentuk pada permukaan luar kristal, tegangan permukaan yang terbentuk akan menghambat laju aliran elektron. Seperti ilustrasi penyerapan O_2 oleh sensor pada gambar di bawah ini:

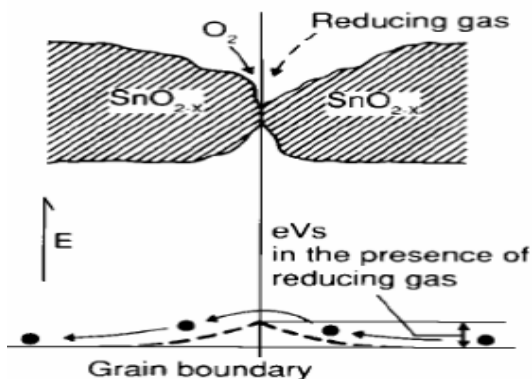


Gambar 2. 1 Ilustrasi penyerapan O_2 oleh sensor [3]

Sensor gas tipe semikonduktor terdiri dari elemen sensor, dasar sensor dan tudung sensor. Elemen sensor menggunakan bahan-bahan

seperti timah(IV)oksida SnO_2 , wolfram (VI) oksida WO_3 , dan lain-lain. Bila suatu kristal oksida logam seperti SnO_2 dipanaskan pada suhu tinggi tertentu di udara, oksigen teradsorpsi pada permukaan kristal dengan muatan negatif. Pada gambar 2.2 elektron – elektron donor pada permukaan kristal ditransfer ke oksigen teradsorpsi, sehingga menghasilkan suatu lapisan ruang bermuatan positif. Akibat dari peristiwa tersebut potensial permukaan terbentuk yang akan menghambat aliran elektron. Proses didalam sensor arus listrik mengalir melalui bagian – bagian penghubung (batas butir) kristal-kristal mikro SnO_2 . Di batas-batas antar butir, oksigen yang teradsorpsi membentuk penghalang potensial yang menghambat muatan bebas bergerak. Tahanan listrik sensor disebabkan oleh penghalang potensial.

Pada gambar 2.2 ketika Model penghalang potensial antar butir dalam lingkungan gas atau terdeteksi gas, kerapatan oksigen teradsorpsi bermuatan negatif pada permukaan semikonduktor sensor menjadi berkurang, sehingga ketinggian penghalang pada batas antar butir berkurang. Sehingga menyebabkan berkurangnya tahanan butir dalam lingkungan gas.



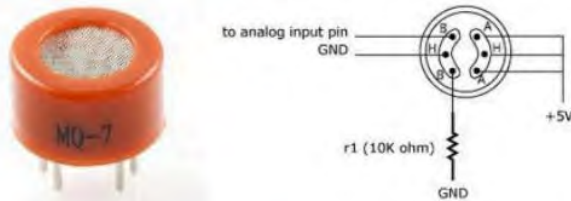
Gambar 2.2 Model penghalang potensial antar butir dalam lingkungan gas[3]

Hubungan antar tahanan sensor dan konsentrasi gas pereduksi pada suatu rentang konsentrasi gas dapat dinyatakan dengan persamaan berikut [3] :

$$R_s = A[C]^{-a} \quad (2.2)$$

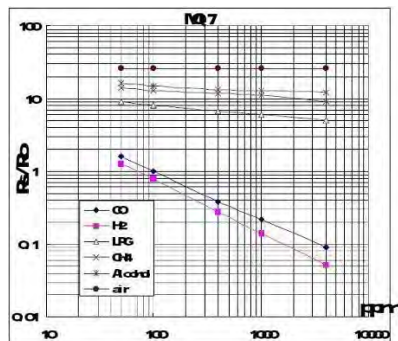
R_s = Resistansi Sensor, A = Konstanta, $[C]$ = Konsentrasi gas, a = gradien kurva R_s

2.2.1 Sensor Gas MQ-7



Gambar 2.3 Sensor Gas MQ-7 [4]

Sensor gas MQ-7 merupakan sensor gas karbon monoksida (CO) yang berfungsi untuk mengetahui konsentrasi gas karbon monoksida (CO). Dimana sensor ini salah satunya dipakai dalam memantau gas karbon monoksida (CO). Sensor ini memiliki sensitivitas tinggi dan waktu respon yang cepat. Keluaran yang dihasilkan oleh sensor ini adalah berupa sinyal analog. Pada gambar 2.3 sensor ini membutuhkan tegangan direct current (DC) sebesar 5V. Pada sensor ini terdapat nilai resistansi sensor (R_s) yang dapat berubah bila terkena gas dan juga sebuah pemanas yang digunakan sebagai pembersihan ruangan sensor dari kontaminasi udara luar. Sensor ini memerlukan rangkaian sederhana serta memerlukan tegangan pemanas (power heater) sebesar 5V, resistansi beban (load resistance), dan output sensor dihubungkan ke analog digital converter (ADC), sehingga keluaran dapat ditampilkan dalam bentuk sinyal digital. Gambar 2.4 menunjukkan sensitivitas sensor MQ-7 [4].



Gambar 2.4 Grafik Sensitivitas Sensor MQ-7

2.2.2 Rangkaian Dasar Pengukuran Sensor Gas

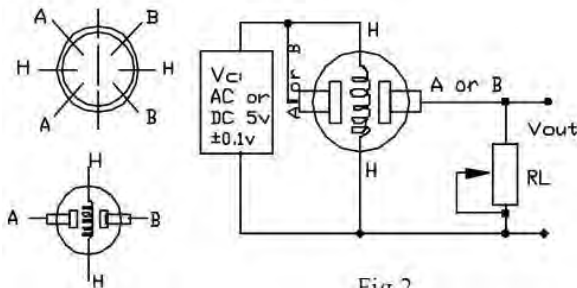


Fig.2

V_H : Voltage pemanas, V_c : Voltage sirkuit Sensor, R_L : Tahanan Beban, V_{RL} : Voltage antar kedua terminal tahanan beban

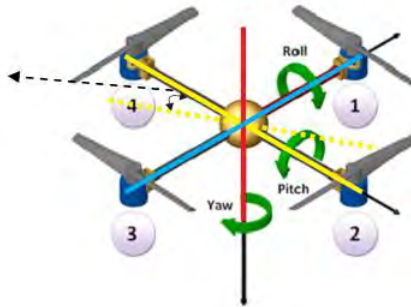
Gambar 2.5 Rangkaian Pengukuran dasar sensor [4]

Sesuai dengan gambar 2.5 sensor memerlukan dua sumber tegangan, yakni tegangan pemanas (V_H) dan tegangan sirkuit/rangkaian sensor (V_c). Tegangan pemanas dipakai pada pemanas terintegrasi untuk mempertahankan elemen sensor pada suhu tertentu yang optimal. Tegangan sirkuit digunakan untuk memungkinkan pengukuran tegangan (V_{RL}) antar kedua terminal tahanan beban (R_L) yang dihubungkan seri dengan sensor. Suatu sirkuit catu daya umum dapat digunakan baik untuk V_c maupun V_H untuk memenuhi kebutuhan listrik sensor.

2.3 Quadcopter

Quadcopter adalah pengembangan dari helicopter yang hanya memiliki sebuah rotor, teknologi yang terdapat pada quadcopter ini menggunakan sinkronisasi antara keempat rotor yang dikonfigurasi dalam bentuk frame plus “+” dimana rotor depan dan belakang berputar searah jarum jam, sedangkan rotor sebelah kanan dan kiri bergerak berlawanan arah jarum jam.

Quadcopter dapat melakukan *take off* dan *landing* secara vertical sehingga biasa disebut sebagai VTOL (*Vertical Take Off and Landing*). Baling-baling depan dan belakang berputar dengan arah yang sama, serta baling-baling kanan dan kiri berputar dengan arah yang sama. Variasi kecepatan rotor, dapat mengubah gaya angkat dan menciptakan gerak. Dengan demikian, meningkatkan atau menurunkan kecepatan keempat baling-baling bersama-sama menghasilkan gerakan vertikal.



Gambar 2.6 Skema Gerakan Quadcopter [5]

Gambar 2.6 menunjukkan bahwa mengubah kecepatan baling-baling 1 dan 3 akan menghasilkan gerakan rotasi roll ditambah dengan gerakan translasi terhadap sumbu y sedangkan mengubah kecepatan baling-baling 2 dan 4 akan menghasilkan gerakan rotasi pitch ditambah dengan gerakan translasi terhadap sumbu x [5].



Gambar 2.7 Quadcopter [5]

Dalam pembuatan *quadcopter*, terdapat beberapa komponen yang menunjang, yaitu :

2.3.1 Remote Control 2.4 GHz

Remote control merupakan bagian yang berinteraksi langsung dengan pengguna untuk memberikan sinyal perintah-perintah untuk menggerakkan robot. Frekuensi yang digunakan sebagai media transmisi adalah gelombang radio pada frekuensi 2,4 Ghz.



Gambar 2.8 Remote Control 2.4 GHz merk Futaba [5]

Remote control memiliki 2 jenis yaitu *single* dan *full function controller*. Pada *remote control* kebanyakan menggunakan *full function controller*, dimana memiliki enam kontrol dasar yaitu : maju, mundur, maju belok kiri, maju belok kanan, mundur belok kiri, mundur belok kanan. Berikut cara kerja *remote control* :

1. Pemain menekan trigger -misal: maju- pada *remote control*. Trigger yang ditekan menghubungkan konektor -saklar- sehingga mengaktifkan pin yang spesifik pada IC.

2. Rangkaian sirkuit aktif yang di-trigger membuat transmitter memancarkan serangkaian set pulsa digital. Setiap rangkaian pulsa memiliki pulsa sinkronisasi yang diikuti dengan segmen pulsa dengan jarak yang berbeda dalam rentang waktu yang konstan. Misal, pulsa sinkronisasi -yang memberitahu receiver akan informasi yang datang- adalah 4 pulsa dengan panjang 2,1 milisecond dan interval 700 microsecond dan pulsa segmen -yang berisi informasi baru- adalah pulsa dengan panjang 700 microsecond dan interval 700 microsecond. Berikut adalah serangkaian pulsa yang digunakan dalam segmen pulsa:

- Maju : 16 pulsa
- Mundur : 40 pulsa
- Maju belok kiri : 28 pulsa
- Maju belok kanan : 34 pulsa
- Mundur belok kiri : 52 pulsa
- Mundur belok kanan : 46 pulsa

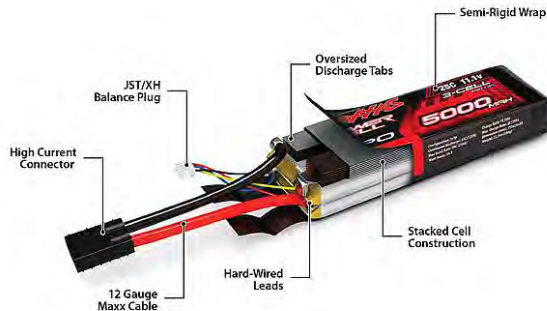
3. Transmitter mengirimkan sinyal pada gelombang radio dengan frekuensi -misal- 2,400,000,000 cycles per second -2,4 GHz-.

4. Receiver memonitor ada tidaknya sinyal pada gelombang 2,4 GHz. Ketika ada sinyal diterima, sinyal kemudian diteruskan pada filter untuk

membuang sinyal yang berada diluar gelombang 2,4 GHz. Sinyal yang ada kemudian kembali dikonversi menjadi serangkaian pulsa elektrik.

5. Serangkaian pulsa itu kemudian diteruskan ke IC yang mendekodekan informasi yang dibawa untuk kemudian diteruskan pada motor. Misal pulsa yang didapat adalah 16 pulsa (maju), maka IC akan memberikan arus positif pada motor sehingga mainan berjalan maju. Jika kemudian didapat pulsa 40 (mundur), maka IC memberikan arus negatif pada motor sehingga mainan berjalan mundur [6].

2.3.2 Baterai Lythium Pollymer (Lipo)



Gambar 2.9 Konfigurasi Baterai Lithium Pollymer [6]

Komponen sumber energi utama dari sebuah *quadcopter* adalah baterai LiPo. Seperti gambar 2.9, baterai lipo tidak menggunakan cairan sebagai elektrolit melainkan menggunakan elektrolit polimer kering yang berbentuk seperti lapisan plastik film tipis. Lapisan film ini disusun berlapis-lapis diantara anoda dan katoda yang mengakibatkan pertukaran ion. Dengan metode ini baterai LiPo dapat dibuat dalam berbagai bentuk dan ukuran. Diluar dari kelebihan arsitektur baterai LiPo, terdapat juga kekurangan yaitu lemahnya aliran pertukaran ion yang terjadi melalui elektrolit polimer kering. Hal ini menyebabkan penurunan pada *charging* dan *discharging rate*.

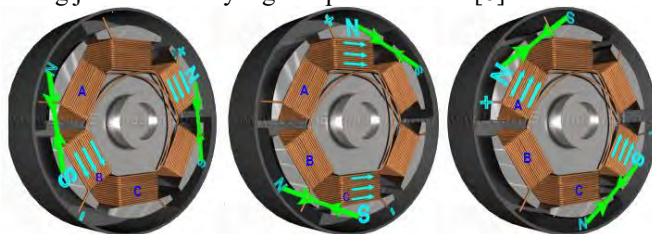
Cara pemilihan baterai adalah berdasarkan jumlah sel dan kapasitasnya. Jumlah sel dalam baterai LiPo adalah sebesar 3,7 Volt. Sehingga bila menggunakan LiPo 3 sel berarti memiliki ukuran 11,1 Volt. Kapasitas baterai LiPo dinyatakan dalam mAh, dimana semakin besar nilai mAh semakin mampu menyimpan tenaga. Semakin besar kapasitas baterai LiPo maka semakin berat ukuran baterai tersebut sehingga akan

menurunkan kemampuan terbang di udara. Umumnya untuk *multirotor* ukuran kecil, menggunakan kapasitas 1500mAh-2200mAh, sedangkan untuk *multirotor* ukuran besar menggunakan kapasitas 4000mAh ke atas [6].

2.3.3 Brushless DC (BLDC) Motor

BLDC motor adalah suatu jenis motor-sinkhron. Artinya medan magnet yang dihasilkan oleh stator dan medan magnet yang dihasilkan oleh Rotor berputar di frekwensi yang sama. Rotor dibuat dari magnet tetap dan dapat desain dari dua sampai delapan kutub magnet. Biasanya magnet ferrit yang dipakai untuk membuat magnet tetap. Sensor tambahan dan elektronik mengontrol amplitudo keluaran inverter dan gelombang. Rotor bagian dari motor brushless sering disebut motor sinkron magnet permanen, tetapi juga bisa menjadi motor induksi. Dua parameter kinerja utama dari *brushless motor DC* adalah motor konstanta KV dan KM.

Gambar 2.10 menunjukkan cara kerja *brushless motor DC*. Pemberian tegangan AC 3 fasa pada stator BLDC adalah menciptakan medan magnet putar stator untuk menarik magnet rotor. Dimana 2 *coil* pertama stator ber kutub utara menarik magnet rotor kutub selatan, dan pada 2 *coil* stator ketiga memberi gaya tolak karena memiliki kutub yang sama. Hal ini berlangsung terus menerus sehingga menyebabkan BLDC motor berputar. Oleh karena tidak adanya *brush* pada motor BLDC, untuk menentukan timing komutasi yang tepat pada motor ini sehingga didapatkan torsi dan kecepatan yang konstan, diperlukan 3 buah sensor Hall atau encoder. Pada sensor hall, *timing* komutasi ditentukan dengan cara mendeteksi medan magnet rotor dengan menggunakan 3 buah sensor hall untuk mendapatkan 6 kombinasi timing yang berbeda, sedangkan pada encoder, timing ditentukan dengan menghitung jumlah kutub yang ada pada encoder [6].



Gambar 2.10 Konfigurasi Putaran Brushless Motor DC [6]

2.3.4 Propeller

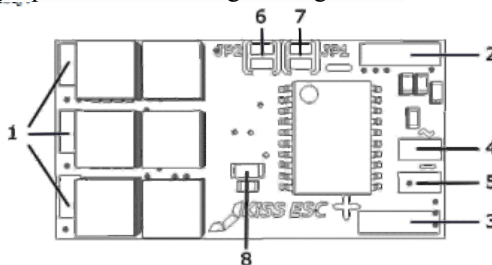


Gambar 2.11 Perbandingan gaya dorong *propeller* 10x8 dengan 10x4

Propeller adalah salah satu bagian mesin yang berfungsi sebagai alat penggerak mekanik, misalnya pada pesawat terbang, kapal laut, hovercraft dan berbagai jenis turbin. Sebuah *quadcopter* menggunakan dua *clock wise* (CW) dan dua *counter clock wise* (CCW) *propeller*. *Propeller* diklasifikasikan berdasarkan panjang diameter dan lebar *pitch*. Misalnya pada *propeller* 9x4,7, *propeller* memiliki panjang 9 inchi dan *pitch* 4,7. Secara umum peningkatan diameter *propeller* akan menarik lebih banyak arus. *Pitch* yang lebih tinggi akan mengakibatkan rotasi semakin lambat, tetapi akan meningkatkan kecepatan *quadcopter* yang juga akan menambah daya lebih seperti pada gambar 2.11 [6].

2.3.5 Electronic Speed Controller (ESC)

ESC menjadi salah satu komponen elektronik yang cukup penting, tugasnya adalah mengubah signal yang diterima untuk memberikan supply listrik yang diperlukan ke masing-masing motor.



1 : konektor ke BLDC, 2 : Konektor lipo negatif, 3 : Konektor lipo positif, 4 : Sinyal input pwm, 5 : *Ground reference* sinyal pwm, 6 : solder *jumper*, 7 : solder *jumper*, 8 : Led.

Gambar 2.12 Konfigurasi *Electronic Speed Controller* [6]

Gambar 2.12 menunjukkan konfigurasi ESC secara umum. Prinsip kerja ESC adalah pertama, pilot mengirim sinyal dari pemancar radio dan sinyal diterima oleh radio penerima di *quadcopter*. Lalu sinyal tersebut diteruskan ke *flight controller* yang akan mengeluarkan sinyal pwm yang proporsional untuk diteruskan ke ESC [6].

2.4 Navigasi Pada *Quadcopter*

Navigasi atau panduan arah merupakan teknik untuk membaca kedudukan (posisi) dan arah benda terhadap kondisi di sekitarnya. Pada umumnya menemukan kedudukan dan posisi dari benda dapat dilakukan dengan melihat beberapa penanda atau ditandai dengan sensor. *Quadcopter* dapat bergerak secara manual maupun otomatis berkat adanya beberapa sensor yang menunjang pada sebuah *flight controller*. IMU (*Inertial Measurement Unit*) dan *Global Positioning System* (GPS) merupakan jenis sensor untuk menunjang navigasi pada *quadcopter*. Dalam perkembangannya, sensor-sensor tersebut terintegrasi menjadi sebuah *flight controller*.

2.4.1 Global Positioning System (GPS)

GPS Adalah navigasi berbasis sistem satelit yang menyediakan informasi lokasi dan waktu. Navigasi GPS terbentuk dari garis pandang empat satelit atau lebih. Setiap sudut-sudut atau daerah di atas permukaan bumi ini minimal terjangkau oleh 3-4 satelit. Pada prakteknya, setiap GPS terbaru bisa menerima sampai dengan 12 channel satelit sekaligus.

Satelit GPS berputar mengelilingi bumi selama 12 jam di dalam orbit yang akurat dan mengirimkan sinyal informasi ke bumi. GPS receiver mengambil informasi itu dan dengan menggunakan perhitungan “*triangulation*” menghitung lokasi user dengan tepat menggunakan *travel time radio*. GPS receiver membandingkan waktu sinyal dikirim dengan waktu sinyal tersebut diterima. Dari informasi itu didapat diketahui berapa jarak satelit. GPS receiver dapat melakukan perhitungan dan menentukan posisi user dan menampilkan dalam peta elektronik.

Konfigurasi pin GPS M8N ditunjukkan pada gambar 2.13. Navigasi GPS memiliki tingkat error ± 5 meter terhadap posisi aktual, dan tidak bertambah sepanjang waktu. Jika dibandingkan dengan navigasi inersial (dead reckoning), yang tingkat error-nya bertambah seiring bertambahnya waktu, untuk jarak jauh navigasi GPS lebih baik karena tingkat error [7].

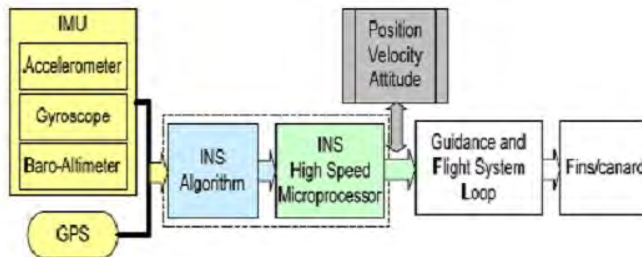


Gambar 2.13 Konfigurasi GPS M8N [7]

2.4.2 Sistem Inertial Measurement Unit (IMU)

Inertial Measurement Unit (IMU) merupakan suatu unit dalam modul elektronik yang mengumpulkan data percepatan angular dan akselerasi linear, yang kemudian dikirim ke unit pemroses utama. IMU terdiri dari kombinasi *accelerometer* (sensor percepatan) dan *gyroscope* (sensor angular) untuk menjejaki keberadaan dan pergerakan suatu benda. *Accelerometer* digunakan untuk mengukur percepatan suatu benda dan *gyroscope* digunakan untuk mengukur rotasi dari suatu benda. Sebuah penelitian terhadap system instrumentasi IMU dapat dilakukan dengan memanfaatkan sensor *accelerometer* dan sensor *gyroscope* [7].

Suatu informasi posisi / jarak dapat diperoleh dari keluaran sensor *accelerometer* melalui proses algoritma tertentu. Sesuai dengan gambar 2.14, *Inertial Measurement Unit* (IMU) adalah komponen utama dalam sistem GNC (*Guidance, Navigation, and Control*) yang berfungsi menangkap parameter-parameter sinyal yang dibutuhkan dalam sistem navigasi.

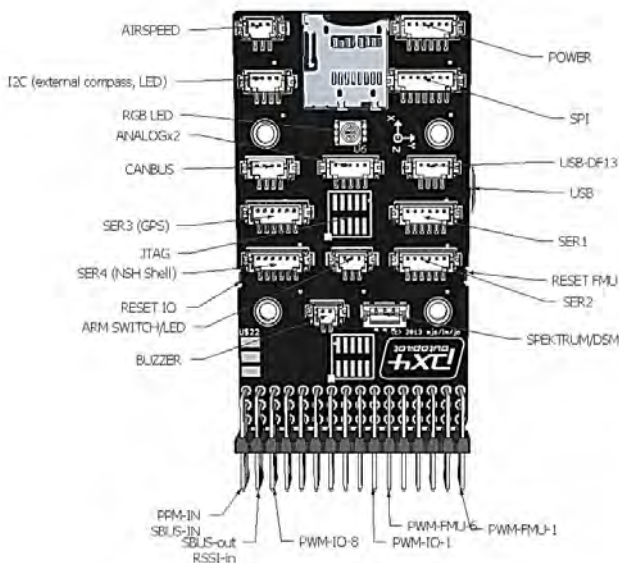


Gambar 2.14 Diagram Blok Sistem GNC [7]

2.4.3 PIXHAWK Flight Controller

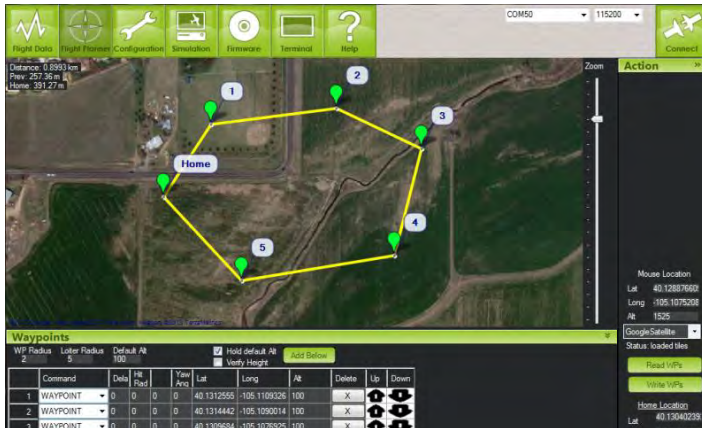
Flight Controller (FC) merupakan otak atau komputer utama dari *multirotor*. Dalam sebuah FC terdapat beberapa *hardware sensor* seperti *Accelerometer*, *Gyro*, *Magnetometer*, *GPS*. Modul *pixhawk* membaca seluruh data sensor dan memproses seluruhnya untuk memerintahkan keempat motor untuk berputar sesuai dengan perintah dari FC. *Pixhawk* merupakan salah satu jenis FC di antara berbagai macam jenis FC. Berikut spesifikasi dari Pixhawk V2.46 PX4 :

- 168 MHz / 252 MIPS Cortex-M4F.
- 14 Output PWM/Servo.
- Beberapa konektivitas untuk *peripheral* tambahan (UART, I2C, CAN).
- Sistem *backup* yang terintegrasi dengan sistem *autopilot* dan manual.
- 32bit STM32F427 Cortex M4 core with FPU.
- ST Micro L3GD20H 16 bit gyroscope.
- ST Micro LSM303D 14 bit accelerometer / magnetometer.
- MEAS MS5611 barometer. [8]



Gambar 2.15 Konfigurasi Pixhawk Flight Controller

2.4.4 Software Mission Planner



Gambar 2.16 Software Mission Planner

Software mission planner digunakan untuk membuat *waypoint* pada *quadcopter* yang sudah terinstal APM/Pixhawk. Pada fokus Tugas Akhir disini, *software mission planner* ini digunakan untuk memberikan titik-titik lokasi (*waypoint*) dimana *Quadcopter* harus menyusurnya.

Fitur – fitur dari *software mission planner* [9]:

- Point-and-click waypoint entry*, menggunakan Google Maps. Dimana pengguna memberikan input berupa titik-titik waypoint yang ingin dituju dan ketinggian yang diinginkan.
- Memilih perintah misi dari menu *drop-down*.
- Mendownload mission log file dan menganalisanya.
- Mengkonfigurasi setting APM.
- Tampilan status *vehicle* dari serial terminal APM.

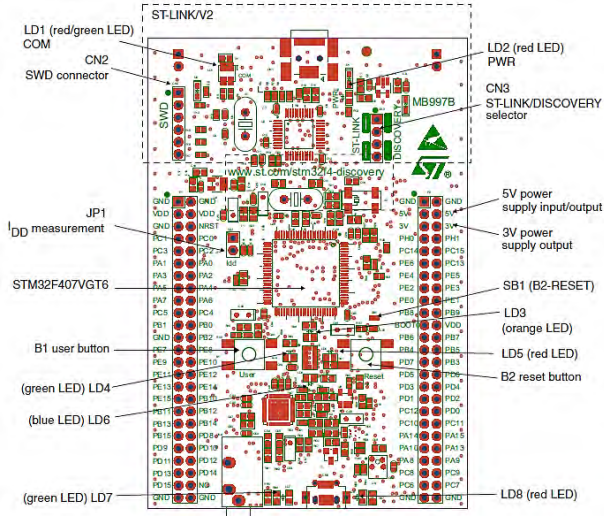
2.5 STM32F4

STM32F4 adalah produk dari ST Electronics yang memudahkan penggunanya untuk melakukan berbagai macam eksperimen elektronik. Bentuk board dari STM32F4 dapat dilihat pada gambar 2.16.

Berikut adalah beberapa fitur dari STM32F4 [10] :

- STM32F407VGT6 *microcontroller* menggunakan 32-bit ARM Cortex-M4F core, 1 MB Flash dan 192 KB RAM
- Supply Board melalui USB atau eksternal 5 V

3. Eksternal power supply: 3 V and 5 V yang disediakan oleh board STM32F4
4. LD1 (merah/hijau) untuk komunikasi dengan USB
5. LD2 (merah) funtuk 3.3 V indikator supply



Gambar 2.17 STM32F4- Discovery [10]

2.6 Raspberry Pi 2

Raspberry Pi, sering juga disingkat dengan nama Raspi, adalah komputer papan tunggal (Single Board Circuit /SBC) yang memiliki ukuran sebesar kartu kredit. Raspberry Pi bisa digunakan untuk berbagai keperluan, seperti spreadsheet, game, bahkan bisa digunakan sebagai media player karena kemampuannya dalam memutar video high definition.

Desain Raspberry Pi didasarkan seputar SoC (System-on-a-chip) Broadcom BCM2835, yang telah menanamkan prosesor ARM1176JZF-S dengan 700 MHz, VideoCore IV GPU, dan 256 Megabyte RAM (model B). Penyimpanan data didesain tidak untuk menggunakan hard disk atau solid-state drive, melainkan mengandalkan kartu SD (SD memory card) untuk booting dan penyimpanan jangka panjang. Raspberry Pi bersifat open source (berbasis Linux), Raspberry Pi bisa dimodifikasi sesuai kebutuhan penggunaannya. Sistem operasi utama Raspberry Pi menggunakan Debian GNU/Linux dan bahasa pemrograman

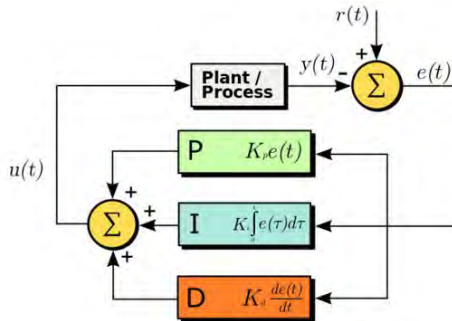
Python. Pada penelitian ini, raspberry pi digunakan sebagai *data logger* berupa web server.



Gambar 2.18 Raspberry Pi 2 Model B

2.7 Pengendali Propotional Integral Deratif (PID)

Untuk menutupi semua kekurangan pengendali PI maupun pengendali PD, maka ketiga mode yang ada digabung menjadi pengendali PID. Unsur P, I, maupun D berfungsi untuk mempercepat reaksi sistem menghilangkan offset. Karena masing-masing mempunyai kelebihan, maka men-tuning K_p , K_i , dan K_d unsur tersebut dibuat lebih menonjol dari pada yang lain. Misalnya untuk P dibuat lebih menonjol dari I maupun D, atau unsur I dibuat lebih menonjol daripada P maupun D, unsur yang menonjol itulah yang kemudian akan membawa pengaruh pada respon sistem keseluruhan.



Gambar 2.19 Implementasi PID pada plant dengan feedback

Dari gambar 2.19 fungsi transfer untuk pengendalian PID adalah :

$$O = K_p * e + K_i \int e \, dt + K_d \, de/dt \quad (2.3)$$

dimana :

$K_p = 100\%/PB$ (konstanta penguatan proporsional)

$K_i = K/T_i$ (konstanta penguatan integral)

$K_d = K_p \times T_d$ (konstanta penguatan diferensial)

Pada tanggapan transien ini ada beberapa parameter yang perlu diketahui, yaitu:

- a. Waktu Tunda (*Delay Time*), adalah waktu yang diperlukan sistem untuk mencapai seperuh dari harga akhirnya untuk pertama kali.
- b. Waktu Naik (*Rise Time*), adalah waktu yang diperlukan sistem untuk naik dari 10% sampai 90% nilai akhir.
- c. Waktu Puncak (*Peak Time*), waktu yang diperlukan sistem untuk mencapai puncak pertama kali.

BAB III

PERANCANGAN SISTEM

Perancangan sistem meliputi perangkat keras dan perangkat lunak. Perangkat keras terbagi meliputi perangkat mekanik *quadcopter* dan elektrik. Perangkat mekanik *quadcopter* dirancang sesuai dengan perangkat elektrik yang akan diaplikasikan pada *quadcopter*. Sehingga *quadcopter* dapat terbang melakukan pemetaan dengan membawa perangkat-perangkat elektrik dan sensor. Perangkat elektrik yang akan digunakan antara lain Pixhawk *Flight Controller*, STM32F4, *GPS + Compass Module*, Raspberry Pi 2, *power bank*, dan *power module*. Sensor yang digunakan merupakan sensor gas MQ-7. Sensor ini merupakan sensor yang sensitif dengan udara terkontaminasi gas Karbon Monoksida (CO).

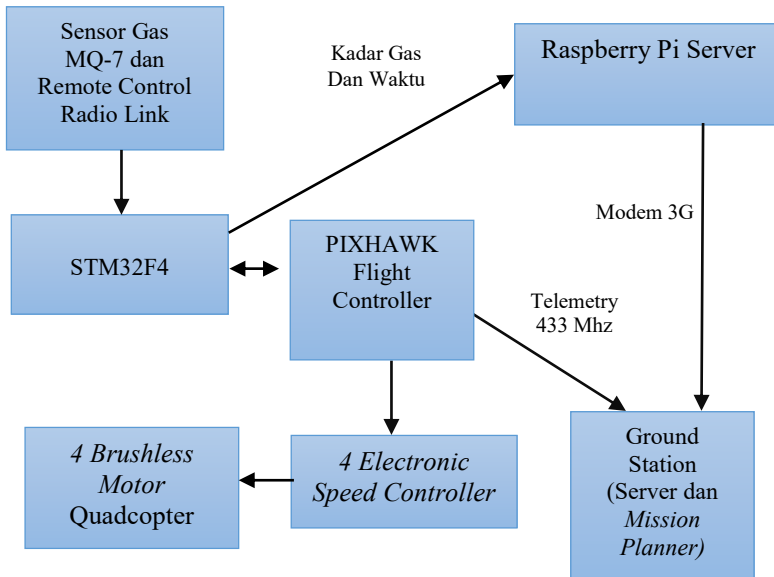
Perangkat lunak meliputi program STM32F4-Discovery menggunakan *software Coocox*, program python Raspberry Pi 2, dan kendali *waypoint* dan kontrol Proporsional Integral Derivatif (PID) menggunakan *software mission planner*. Kendali *waypoint* adalah sistem yang digunakan oleh *quadcopter* untuk melakukan pemetaan gas polutan secara otomatis. Kontrol Proporsional Integral Derivatif (PID) adalah sistem kontrol yang digunakan untuk mengkontrol keseimbangan *quadcopter*. Bab ini menjelaskan secara keseluruhan dan desain tiap – tiap blok yang ada. Perangkat sistem ini didesain dengan sumber baterai lipo 4 sel 5200 mah 14,8 volt untuk *electric speed controller* dan *brushless motor*, 5 volt untuk STM32-F4 Discovery dan *Driver Sensor*.

3.1 Diagram Blok Sistem

Secara umum sistem ini terdiri dari perangkat keras dan perangkat lunak. Perangkat keras meliputi perancangan perangkat elektrik dan perangkat mekanik. Perangkat lunak meliputi program STM32F4, Raspberry Pi 2, kendali *waypoint*, dan kontrol proporsional integral derivatif (PID) menggunakan *software mission planner*.

Diagram blok sistem alat yang dirancang pada tugas akhir ini ditunjukkan pada gambar 3.1. Dimana sistem kerja dari *quadcopter* adalah mendeteksi dan menjangkau sumber gas dengan menggunakan sensor semikonduktor MQ-7 dengan metode *waypoint*. Proses interface yang digunakan untuk menghubungkan ke komputer adalah menggunakan raspberry pi 2 sebagai *web server online* dengan modem

sebagai perantara. Data berupa kadar PPM dapat ditampilkan pada suatu *web socket* bernama *plotly*, dimana data disajikan dalam grafik secara *online real time*.



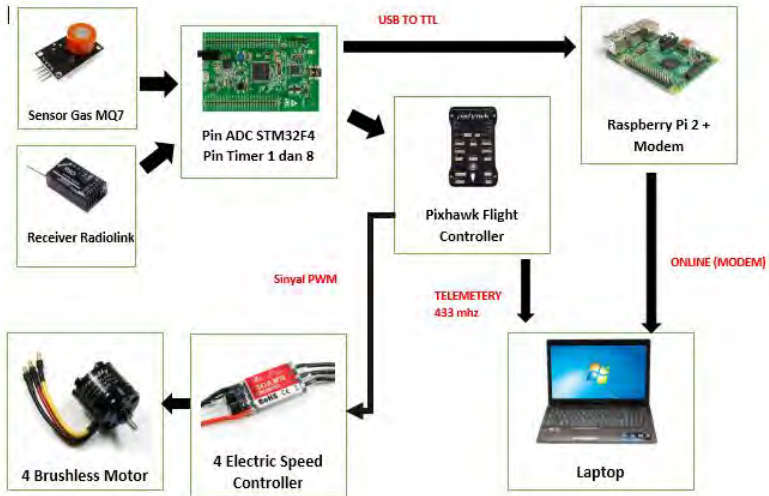
Gambar 3.1 Diagram Blok Sistem Quadcopter

Software *Mission Planner* yang terhubung dengan *telemetry* 433 Mhz menampilkan data pemetaan *waypoint* dari *quadcopter*. Kedua data tersebut terhubung secara langsung pada sebuah *ground station* berupa laptop.

Output dari sensor berupa tegangan dalam wujud data analog yang diinputkan ke STM32F4. Sebelum diinputkan ke web server raspberry pi, data analog ini harus diubah dalam bentuk digital dengan memanfaatkan ADC (*Analog Digital Converter*) internal dari STM32F4 dengan resolusi ADC 10 bit. Hasil ADC berupa data tegangan ADC yang akan dikonversi menjadi nilai ppm (*part per million*). Nilai kadar gas PPM tersebut dikirim melalui komunikasi serial ke raspberry pi 2 menggunakan USB to TTL. Di dalam raspberry pi 2, data ppm diolah menggunakan program python sehingga data mampu disajikan ke web server *plotly*.

Sistem kendali *quadcopter* menggunakan *remote control* Radiolink-AT 9 dan *pixhawk flight controller*. Sinyal PWM dari *remote control*

dibaca oleh *receiver* radiolink dan diinputkan ke dalam pin timer 1 dan 8 STM32F4. Hasil pembacaan sinyal PWM masing-masing channel remote diolah dan diteruskan ke pixhawk *flight controller*. Kendali auto maupun manual bisa diubah dengan mengubah nilai pwm pada channel 5.



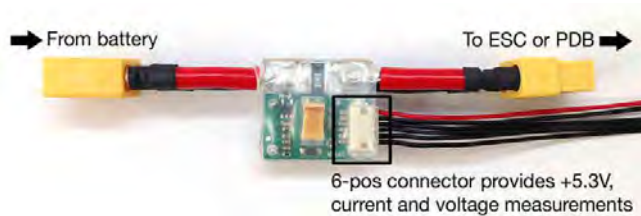
Gambar 3.2 Komponen Sistem Quadcopter

Ketika mode *quadcopter* dideklarasikan, maka pixhawk flight controller akan mengolah input pwm dari STM32F4 untuk mengeluarkan sinyal pwm ke 4 *electric speed controller* (ESC). 4 ESC ini akan mengolah sinyal pwm tersebut ke 4 *brushless motor DC* (BLDC). Sehingga *quadcopter* mampu bergerak sesuai dengan titik *waypoint* yang diinginkan. Komponen-komponen yang akan digunakan sesuai dengan gambar 3.2.

3.2 Perancangan Perangkat Elektrik

Perangkat elektrik meliputi *power module distributor*, *power supply*, rangkaian sensor, *electronic nose*, STM32F4, pixhawk, dan raspberry pi 2.

3.2.1 Power Module Distributor

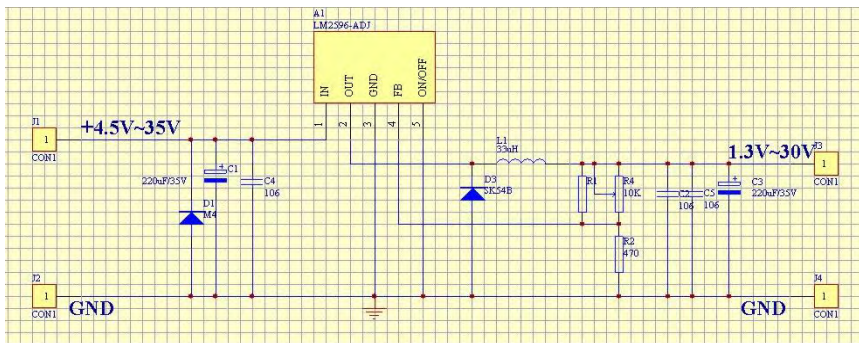


Gambar 3.3 Power Module Pixhawk [7]

Power module distributor merupakan *board* untuk menyalurkan suplai tegangan ke 4 *electric speed controller* dan pixhawk. Untuk pixhawk ditambahkan sebuah rangkaian *power module* tersendiri agar dapat mengonversi tegangan dari baterai lipo 4 sel ke 5 volt.

3.2.2 Power Supply

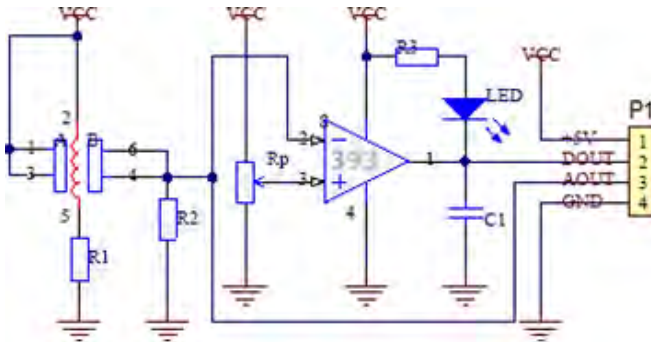
Power supply terdiri dari rangkaian *buck converter* dan rangkaian 12 volt. *Buck converter* berguna untuk mengubah tegangan 12 volt dari baterai lipo 3 sel menjadi 5 volt. Output 5 volt ini digunakan pada STM32F4 dan rangkaian sensor MQ-7. Sedangkan rangkaian 12 volt untuk memberi suplai ke kipas sensor gas.



Gambar 3.4 Rangkaian Power Supply dan *Buck Converter*[8]

3.2.3 Rangkaian Sensor Gas MQ-7

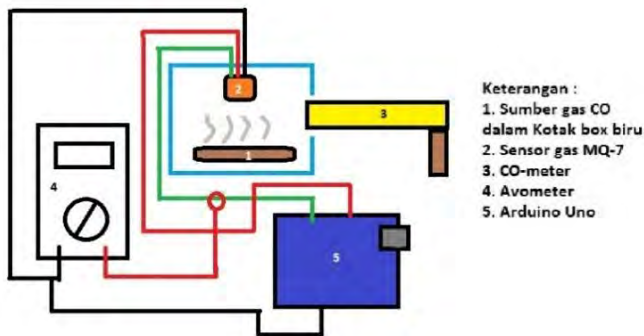
Sensor gas MQ-7 merupakan sensor gas semikonduktor yang dapat mendeteksi gas CO (Karbon Monoksida) di udara.



Gambar 3.5 Rangkaian Sensor MQ-7

Agar didapatkan data tegangan output sensor, maka menggunakan modul rangkaian sensor MQ-7 yang telah ada di pasaran. Sesuai dengan gambar 3.5, sensor gas MQ-7 diberi suplai tegangan sebesar 5 volt untuk bagian *heating* dan *sensing*. Prinsip sensor gas menggunakan *voltage divider*, dimana output dari resistansi *sensing* MQ-7 digabungkan dengan rangkaian beban sebesar 10 Kohm. Sehingga didapatkan hasil tegangan output analog.

Supaya sensor gas MQ-7 dapat mengkonversi hasil tegangan output menjadi satuan *part per million* (ppm) maka diperlukan proses kalibrasi. Proses kalibrasi ini menggunakan arduino uno dan CO-meter.



Gambar 3.6 Konfigurasi kalibrasi sensor gas MQ-7

Arduino uno membaca data berupa tegangan ADC dari sensor dan CO-meter menunjukkan data kadar gas dalam ppm pada wadah tertutup. Kedua data ini kemudian dibandingkan untuk mendapatkan model

matematis. Konfigurasi dari gambar 3.6 adalah pin A0 arduino uno disambungkan ke pin tegangan output dari sensor gas MQ-7. Pin tegangan output sensor gas juga disambungkan ke avometer. Hal ini untuk mendapatkan data tegangan output di avometer dan di pin ADC arduino uno.

3.2.4 STM32F4

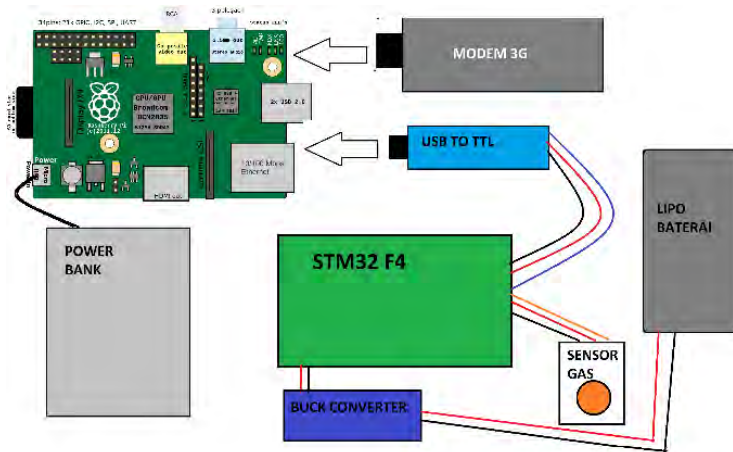
Dalam tugas akhir ini menggunakan STM32F4. Tegangan *Output* pada sensor berupa data analog. Fungsi dari STM32F4 ini untuk mengkonversi data *analog* menjadi *digital* dengan memanfaatkan ADC *internal* dengan resolusi 10 bit dan mengolahnya menjadi *input* untuk dikonversi pada sebuah algoritma agar menghasilkan data PPM (*part per million*).

Algoritma konversi tersebut didapatkan dari hasil kalibrasi sensor gas MQ-7 dengan gas CO. STM32F4 membutuhkan suplai daya dengan tegangan tetap 5V seperti pada rangkaian *power supply* STM32F4 pada gambar 4. Dalam tugas akhir ini, tegangan 5 volt dihasilkan oleh rangkaian *buck converter*.

Proses pemrograman menggunakan software *Coccox* Proses *download* program dari komputer ke STM32F4 cukup menggunakan USB to miniUSB. Sensor yang digunakan berjumlah satu sensor MQ-7, dan menggunakan port A0. Tombol *reset* pada STM32F4 dapat digunakan untuk mereset program di dalam STM32F4. STM32F4 telah dilengkapi *USART* untuk proses pengiriman dan penerima data. Keluaran STM32F4 ini dikirim ke raspberry pi 2 dengan menggunakan *port serial TX (transmitter)* pada port A2 dan A3.

Proses pengolahan sinyal PWM *remote control* menggunakan fitur timer pada STM32F4. Pada tugas akhir ini, *remote control* radiolink memiliki 9 channel, namun yang digunakan hanyalah 8 channel saja. Sehingga input berupa receiver dimasukkan ke timer 3 dan timer 4. Output hasil pengolahan sinyal pwm dikeluarkan pada timer 1 dan 9. Lalu disambungkan ke *ppm encoder* sebelum diolah ke pixhawk.

3.2.5 Raspberry Pi 2



Gambar 3.7 Konfigurasi Raspberry Pi 2

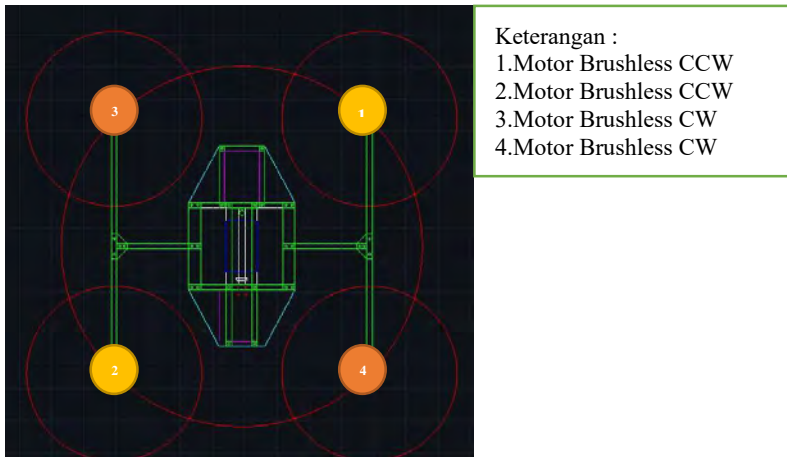
Raspberry pi 2 berguna sebagai web server untuk menyajikan data kadar gas PPM dalam sebuah web socket dan memiliki konfigurasi seperti gambar 3.7. Raspberry pi 2 memiliki input tegangan sebesar 5 volt 1,8 A. Agar dapat memenuhi suplai tegangan, maka sebuah *power bank* 5000mah dipasang pada *quadcopter*. USB to TTL berisi data kadar gas dari STM32F4 dimasukkan pada port USB0. Modem untuk akses online ke web socket dipasang pada port USB3.

3.2.6 Pixhawk

Pixhawk merupakan *flight controller* berbasis ARM STM32 yang dapat digunakan input-output hingga 8 channel. Konfigurasi pixhawk pada tugas akhir ini ditunjukkan pada gambar 3.8. Dalam tugas akhir ini, pixhawk digunakan untuk menerima input dari STM32F4 lalu meneruskan data sinyal pwm ke output channel 1 hingga 4. Data GPS dan kompas disambungkan ke port GPS dan I2C. Telemetry 433 Mhz diinputkan pada port telem1 yang berbasis USART. Port power merupakan output tegangan dari *power module* yang telah dikonversi menjadi 5 volt. Terdapat switch untuk *safety* sebelum melakukan *arming* ke *quadcopter*. Buzzer untuk indikator masing-masing perintah.

3.3.2 Desain Awal *Quadcopter*

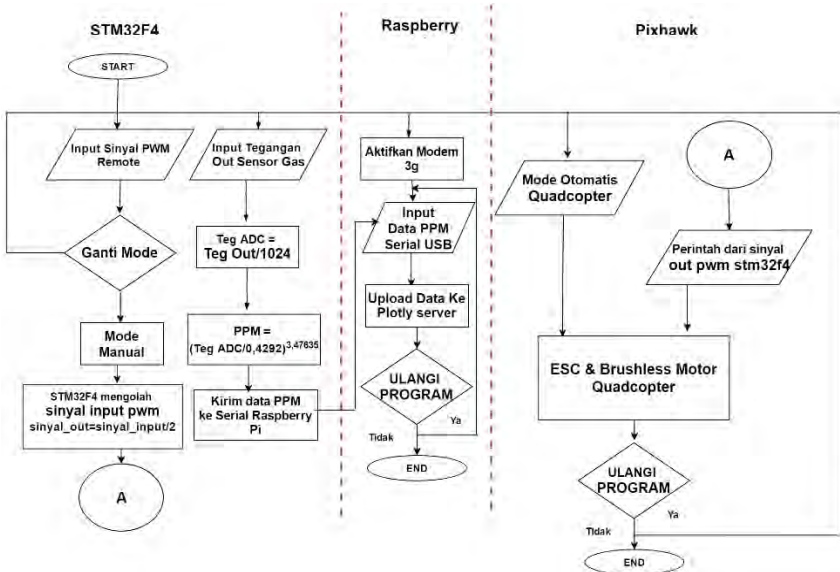
Pada subbab 3.3.1 telah diperhitungkan estimasi beban yang akan dibawa oleh *quadcopter*. Dalam desain awal *quadcopter*, rangka yang akan diterapgunakan menggunakan ukuran s700 (diagonal antar motor 1 dan 3 sepanjang 70cm). Terdapat 4 *brushless motor* DC yang dipasang pada rangka sesiao dengan gambar 3.9. Perancangan ukuran total *quadcopter* pada tugas akhir ini (panjang * lebar) 55x46 cm.



Gambar 3.9 Desain Quadcopter pada Auto Cad

3.4 Perangkat lunak

Perangkat lunak yang dirancang terdiri atas tiga bagian seperti pada gambar 3.10 yaitu perangkat lunak pada STM32F4, perangkat lunak pada pixhawk menggunakan *mission planner*, dan perangkat lunak pada raspberry pi 2. Perangkat lunak pada STM32F4 berupa perangkat lunak untuk melakukan proses ADC sebagai konversi data sensor analog menjadi data PPM dan pemrosesan input berupa sinyal PWM dari *remote control*. Untuk *mission planner*, merupakan proses deklarasi titik *waypoint* dan kontrol *quadcopter* di pixhawk. Sedangkan raspberry pi 2 berguna sebagai web server berbasis *web socket* sehingga dapat menampilkan grafik data ppm secara *online real time*.



Gambar 3.10 Blok sistem perangkat lunak *quadcopter*

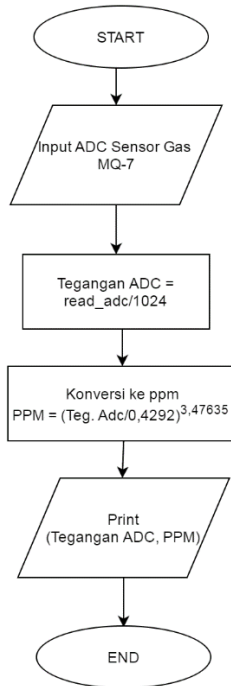
3.4.1 Perangkat Lunak STM32F4

Perangkat lunak STM32 secara umum digunakan untuk mengkonversikan data *analog* menjadi data *digital* dengan memanfaatkan ADC internal. Proses pengolahan sinyal PWM *remote control* menggunakan pin timer. Proses ini menggunakan *software coocox* dalam STM32F4 menggunakan clock 12 MHz, *USART transmitter* dengan *Baud Rate* 9600, pin timer 1, 3, 4, 8, dan ADC yang digunakan 10 bit.

3.4.1.1 Perangkat Lunak ADC STM32F4

Dalam tugas akhir ini menggunakan ADC dengan fidelitas 10 bit dan alur pembacaan ADC sesuai dengan gambar 3.11. Perhitungan tegangan ADC (*Analog Digital Converter*) dapat dirumuskan pada persamaan 3.1 :

$$\text{Tegangan ADC} = \frac{V_{in} \times 1024}{V_{ref}} \quad (3.1)$$



Gambar 3.11 Flowchart Program ADC STM32F4

Dan dalam aplikasi program pembacaan data sensor adalah sebagai berikut:

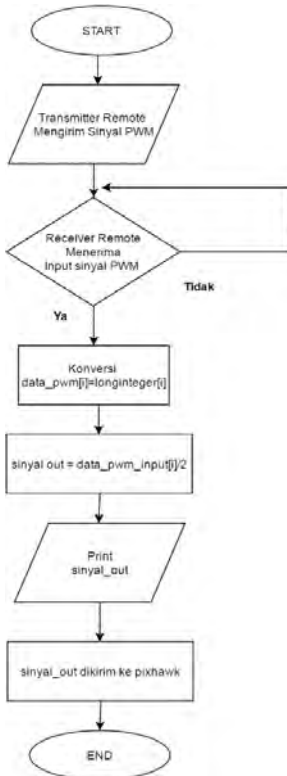
```

for(int x=0; x<100;x++)
{
    sensor_value1 = sensor_value1 + hasil_konversi_adc[0];
}
sensor_value1 = sensor_value1/100.0;

sensor_volt1 = sensor_value1/1024*4.9;
  
```

Dari program diatas data tersebut port yang digunakan adalah Port A0. Lalu program dikonversikan menggunakan algoritma dari hasil kalibrasi sensor gas MQ-7 pada gas karbon monoksida yang telah diketahui.

3.4.1.2 Perangkat Lunak Remote Control STM32F4



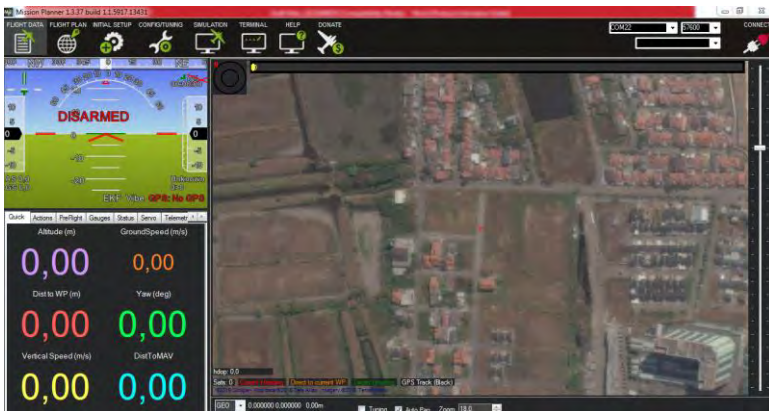
Gambar 3.12 Flowchart Program *Remote Control*

Pada tugas akhir ini digunakan *Remote control* radiolink AT-9 sebagai pengontrol *quadcopter*. Supaya data sinyal PWM *remote control* dapat dibaca oleh pixhawk, maka digunakan pin timer oleh STM32F4 untuk mengolah data sinyal PWM *remote control*.

Sesuai gambar 3.12 program *remote control* dibagi menjadi 2, yaitu saat mengolah input sinyal PWM remote dan mengeluarkan output sinyal PWM hasil pengolahan di STM32F4. Berikut merupakan untuk olah input sinyal PWM pada timer input (pin timer 1 dan 8), sehingga menghasilkan sinyal output PWM pada timer output (pin timer 3 dan 4).

3.4.2 Perangkat Lunak *Mission Planner*

Mission planner merupakan software untuk menunjang controller *quadcopter*, UAV, dan drone berupa *ardupilot* atau *pixhawk*. Pada tugas akhir ini, *mission planner* berperan sebagai penunjang controller *pixhawk*. Deklarasi titik-titik *waypoint* untuk melakukan misi terbang secara otomatis, pemetaan, dapat dilakukan melalui software ini. Kalibrasi sensor-sensor IMU, GPS, dan kompas pada *pixhawk* juga dilakukan melalui *mission planner*.



Gambar 3.13 Tampilan awal software mission planner

Pada gambar 3.13, untuk deklarasi misi dapat dilakukan pada tab *flight plan*, lalu pada bagian perintah *waypoint* di kolom bawah pilih “*add below*”. Maka titik-titik *waypoint* dapat dideklarasikan, dengan cara mengklik ke gambar peta yang tertera. Dalam menjalankan misi, *quadcopter* harus mendapatkan nilai GPS yang sesuai pada posisi *waypoint* yang akan dilaksanakan. Jika tidak, maka *quadcopter* akan *fly away* atau terbang dengan sendirinya tanpa perintah. Setelah titik-titik *waypoint* dideklarasikan, program dimasukkan ke *pixhawk* melalui *telemetry* 433 MHz dengan menekan tombol *Write*.

BAB III

PERANCANGAN SISTEM

Perancangan sistem meliputi perangkat keras dan perangkat lunak. Perangkat keras terbagi meliputi perangkat mekanik *quadcopter* dan elektrik. Perangkat mekanik *quadcopter* dirancang sesuai dengan perangkat elektrik yang akan diaplikasikan pada *quadcopter*. Sehingga *quadcopter* dapat terbang melakukan pemetaan dengan membawa perangkat-perangkat elektrik dan sensor. Perangkat elektrik yang akan digunakan antara lain Pixhawk *Flight Controller*, STM32F4, *GPS + Compass Module*, Raspberry Pi 2, *power bank*, dan *power module*. Sensor yang digunakan merupakan sensor gas MQ-7. Sensor ini merupakan sensor yang sensitif dengan udara terkontaminasi gas Karbon Monoksida (CO).

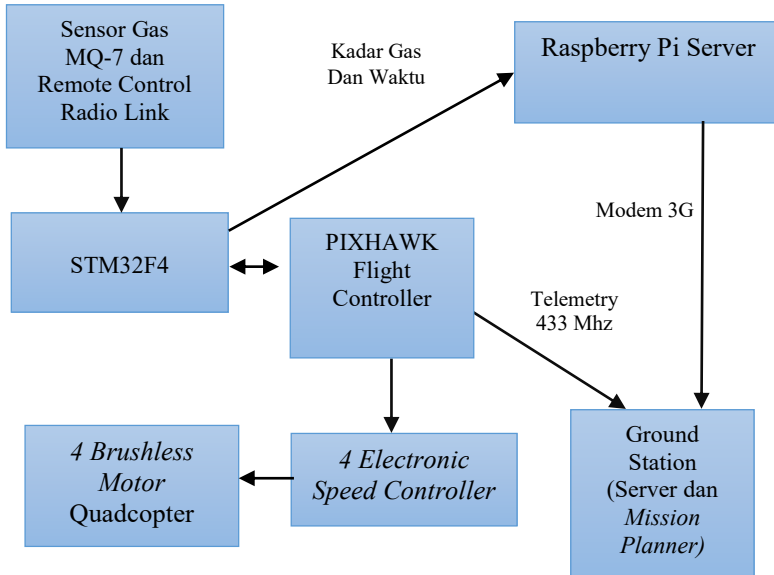
Perangkat lunak meliputi program STM32F4-Discovery menggunakan *software Coocox*, program python Raspberry Pi 2, dan kendali *waypoint* dan kontrol Proporsional Integral Derivatif (PID) menggunakan *software mission planner*. Kendali *waypoint* adalah sistem yang digunakan oleh *quadcopter* untuk melakukan pemetaan gas polutan secara otomatis. Kontrol Proporsional Integral Derivatif (PID) adalah sistem kontrol yang digunakan untuk mengkontrol keseimbangan *quadcopter*. Bab ini menjelaskan secara keseluruhan dan desain tiap – tiap blok yang ada. Perangkat sistem ini didesain dengan sumber baterai lipo 4 sel 5200 mah 14,8 volt untuk *electric speed controller* dan *brushless motor*, 5 volt untuk STM32-F4 Discovery dan *Driver Sensor*.

3.1 Diagram Blok Sistem

Secara umum sistem ini terdiri dari perangkat keras dan perangkat lunak. Perangkat keras meliputi perancangan perangkat elektrik dan perangkat mekanik. Perangkat lunak meliputi program STM32F4, Raspberry Pi 2, kendali *waypoint*, dan kontrol proporsional integral derivatif (PID) menggunakan *software mission planner*.

Diagram blok sistem alat yang dirancang pada tugas akhir ini ditunjukkan pada gambar 3.1. Dimana sistem kerja dari *quadcopter* adalah mendeteksi dan menjangkau sumber gas dengan menggunakan sensor semikonduktor MQ-7 dengan metode *waypoint*. Proses interface yang digunakan untuk menghubungkan ke komputer adalah menggunakan raspberry pi 2 sebagai *web server online* dengan modem

sebagai perantara. Data berupa kadar PPM dapat ditampilkan pada suatu *web socket* bernama *plotly*, dimana data disajikan dalam grafik secara *online real time*.



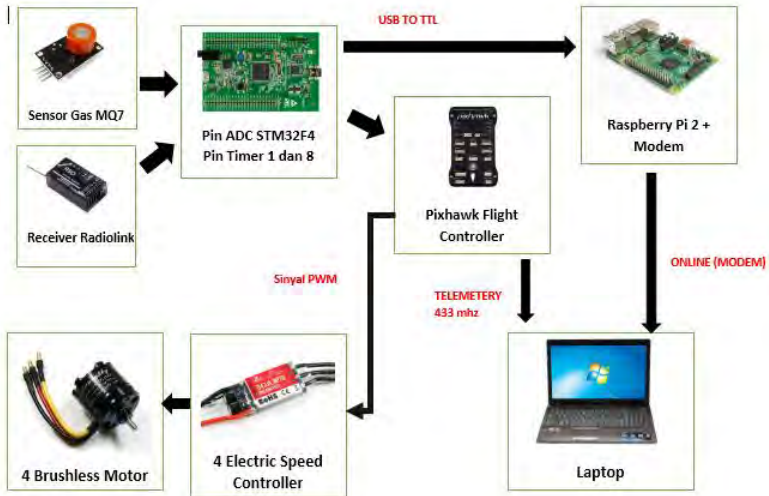
Gambar 3.1 Diagram Blok Sistem Quadcopter

Software *Mission Planner* yang terhubung dengan *telemetry* 433 Mhz menampilkan data pemetaan *waypoint* dari *quadcopter*. Kedua data tersebut terhubung secara langsung pada sebuah *ground station* berupa laptop.

Output dari sensor berupa tegangan dalam wujud data analog yang diinputkan ke STM32F4. Sebelum diinputkan ke web server raspberry pi, data analog ini harus diubah dalam bentuk digital dengan memanfaatkan ADC (*Analog Digital Converter*) internal dari STM32F4 dengan resolusi ADC 10 bit. Hasil ADC berupa data tegangan ADC yang akan dikonversi menjadi nilai ppm (*part per million*). Nilai kadar gas PPM tersebut dikirim melalui komunikasi serial ke raspberry pi 2 menggunakan USB to TTL. Di dalam raspberry pi 2, data ppm diolah menggunakan program python sehingga data mampu disajikan ke web server *plotly*.

Sistem kendali *quadcopter* menggunakan *remote control* Radiolink-AT 9 dan *pixhawk flight controller*. Sinyal PWM dari *remote control*

dibaca oleh *receiver* radiolink dan diinputkan ke dalam pin timer 1 dan 8 STM32F4. Hasil pembacaan sinyal PWM masing-masing channel remote diolah dan diteruskan ke pixhawk *flight controller*. Kendali auto maupun manual bisa diubah dengan mengubah nilai pwm pada channel 5.



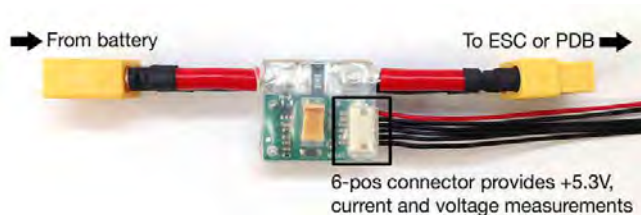
Gambar 3.2 Komponen Sistem Quadcopter

Ketika mode *quadcopter* dideklarasikan, maka pixhawk flight controller akan mengolah input pwm dari STM32F4 untuk mengeluarkan sinyal pwm ke 4 *electric speed controller* (ESC). 4 ESC ini akan mengolah sinyal pwm tersebut ke 4 *brushless motor DC* (BLDC). Sehingga *quadcopter* mampu bergerak sesuai dengan titik *waypoint* yang diinginkan. Komponen-komponen yang akan digunakan sesuai dengan gambar 3.2.

3.2 Perancangan Perangkat Elektrik

Perangkat elektrik meliputi *power module distributor*, *power supply*, rangkaian sensor, *electronic nose*, STM32F4, pixhawk, dan raspberry pi 2.

3.2.1 Power Module Distributor

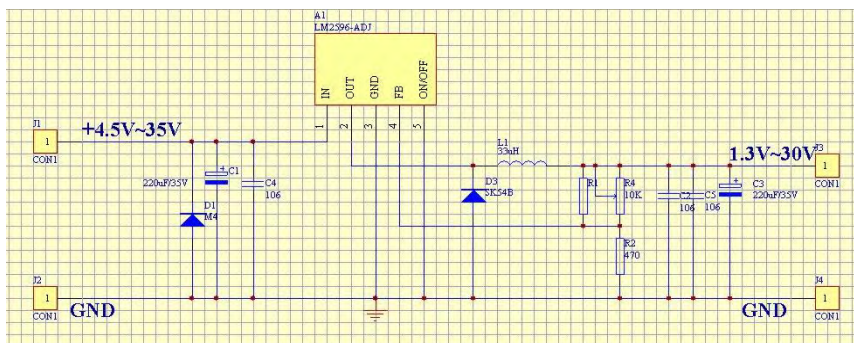


Gambar 3.3 Power Module Pixhawk [7]

Power module distributor merupakan *board* untuk menyalurkan suplai tegangan ke 4 *electric speed controller* dan *pixhawk*. Untuk *pixhawk* ditambahkan sebuah rangkaian *power module* tersendiri agar dapat mengonversi tegangan dari baterai lipo 4 sel ke 5 volt.

3.2.2 Power Supply

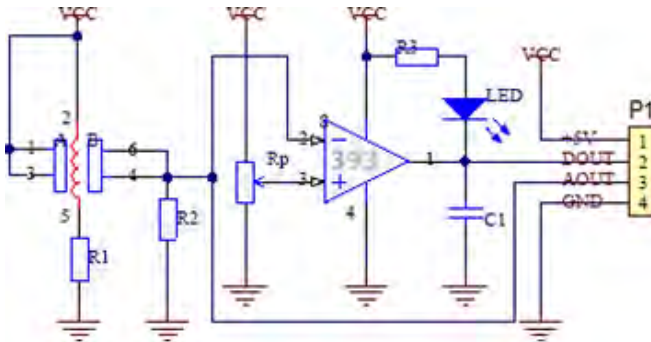
Power supply terdiri dari rangkaian *buck converter* dan rangkaian 12 volt. *Buck converter* berguna untuk mengubah tegangan 12 volt dari baterai lipo 3 sel menjadi 5 volt. Output 5 volt ini digunakan pada STM32F4 dan rangkaian sensor MQ-7. Sedangkan rangkaian 12 volt untuk memberi suplai ke kipas sensor gas.



Gambar 3.4 Rangkaian Power Supply dan *Buck Converter*[8]

3.2.3 Rangkaian Sensor Gas MO-7

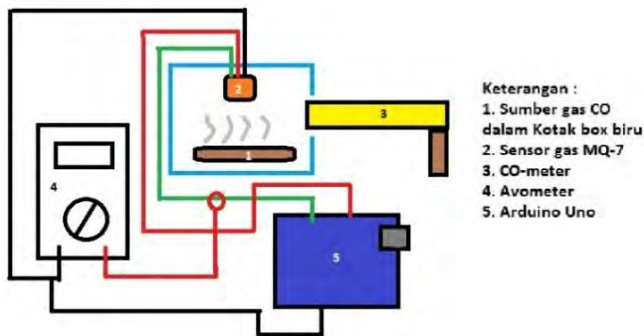
Sensor gas MQ-7 merupakan sensor gas semikonduktor yang dapat mendeteksi gas CO (Karbon Monoksida) di udara.



Gambar 3.5 Rangkaian Sensor MQ-7

Agar didapatkan data tegangan output sensor, maka menggunakan modul rangkaian sensor MQ-7 yang telah ada di pasaran. Sesuai dengan gambar 3.5, sensor gas MQ-7 diberi suplai tegangan sebesar 5 volt untuk bagian *heating* dan *sensing*. Prinsip sensor gas menggunakan *voltage divider*, dimana output dari resistansi *sensing* MQ-7 digabungkan dengan rangkaian beban sebesar 10 Kohm. Sehingga didapatkan hasil tegangan output analog.

Supaya sensor gas MQ-7 dapat mengkonversi hasil tegangan output menjadi satuan *part per million* (ppm) maka diperlukan proses kalibrasi. Proses kalibrasi ini menggunakan arduino uno dan CO-meter.



Gambar 3.6 Konfigurasi kalibrasi sensor gas MQ-7

Arduino uno membaca data berupa tegangan ADC dari sensor dan CO-meter menunjukkan data kadar gas dalam ppm pada wadah tertutup. Kedua data ini kemudian dibandingkan untuk mendapatkan model

matematis. Konfigurasi dari gambar 3.6 adalah pin A0 arduino uno disambungkan ke pin tegangan output dari sensor gas MQ-7. Pin tegangan output sensor gas juga disambungkan ke avometer. Hal ini untuk mendapatkan data tegangan output di avometer dan di pin ADC arduino uno.

3.2.4 STM32F4

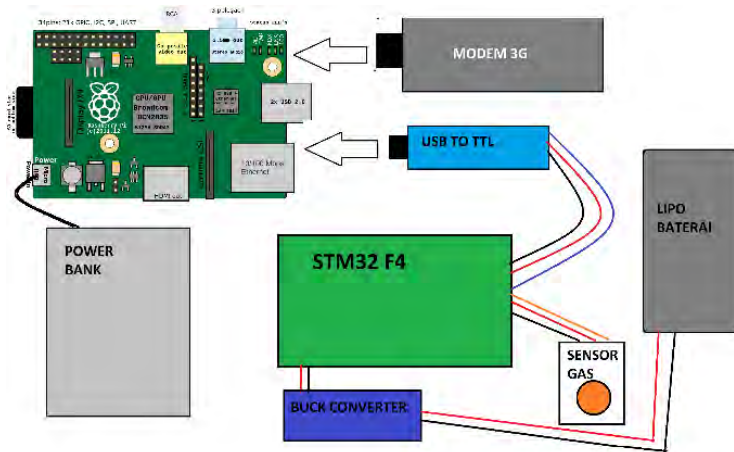
Dalam tugas akhir ini menggunakan STM32F4. Tegangan *Output* pada sensor berupa data analog. Fungsi dari STM32F4 ini untuk mengkonversi data *analog* menjadi *digital* dengan memanfaatkan ADC *internal* dengan resolusi 10 bit dan mengolahnya menjadi *input* untuk dikonversi pada sebuah algoritma agar menghasilkan data PPM (*part per million*).

Algoritma konversi tersebut didapatkan dari hasil kalibrasi sensor gas MQ-7 dengan gas CO. STM32F4 membutuhkan suplai daya dengan tegangan tetap 5V seperti pada rangkaian *power supply* STM32F4 pada gambar 4. Dalam tugas akhir ini, tegangan 5 volt dihasilkan oleh rangkaian *buck converter*.

Proses pemrograman menggunakan software *Coccox* Proses *download* program dari komputer ke STM32F4 cukup menggunakan USB to miniUSB. Sensor yang digunakan berjumlah satu sensor MQ-7, dan menggunakan port A0. Tombol *reset* pada STM32F4 dapat digunakan untuk mereset program di dalam STM32F4. STM32F4 telah dilengkapi *USART* untuk proses pengiriman dan penerima data. Keluaran STM32F4 ini dikirim ke raspberry pi 2 dengan menggunakan *port serial TX (transmitter)* pada port A2 dan A3.

Proses pengolahan sinyal PWM *remote control* menggunakan fitur timer pada STM32F4. Pada tugas akhir ini, *remote control* radiolink memiliki 9 channel, namun yang digunakan hanyalah 8 channel saja. Sehingga input berupa receiver dimasukkan ke timer 3 dan timer 4. Output hasil pengolahan sinyal pwm dikeluarkan pada timer 1 dan 9. Lalu disambungkan ke *ppm encoder* sebelum diolah ke pixhawk.

3.2.5 Raspberry Pi 2

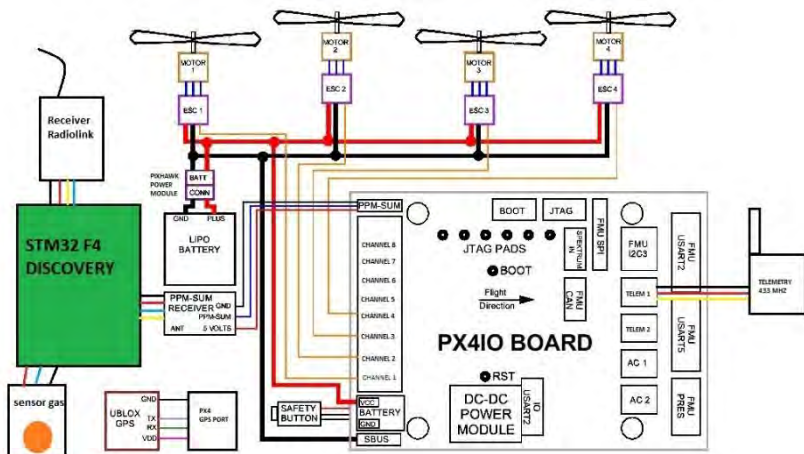


Gambar 3.7 Konfigurasi Raspberry Pi 2

Raspberry pi 2 berguna sebagai web server untuk menyajikan data kadar gas PPM dalam sebuah web socket dan memiliki konfigurasi seperti gambar 3.7. Raspberry pi 2 memiliki input tegangan sebesar 5 volt 1,8 A. Agar dapat memenuhi suplai tegangan, maka sebuah *power bank* 5000mah dipasang pada *quadcopter*. USB to TTL berisi data kadar gas dari STM32F4 dimasukkan pada port USB0. Modem untuk akses online ke web socket dipasang pada port USB3.

3.2.6 Pixhawk

Pixhawk merupakan *flight controller* berbasis ARM STM32 yang dapat digunakan input-output hingga 8 channel. Konfigurasi pixhawk pada tugas akhir ini ditunjukkan pada gambar 3.8. Dalam tugas akhir ini, pixhawk digunakan untuk menerima input dari STM32F4 lalu meneruskan data sinyal pwm ke output channel 1 hingga 4. Data GPS dan kompas disambungkan ke port GPS dan I2C. Telemetry 433 Mhz diinputkan pada port telem1 yang berbasis USART. Port power merupakan output tegangan dari *power module* yang telah dikonversi menjadi 5 volt. Terdapat switch untuk *safety* sebelum melakukan *arming* ke *quadcopter*. Buzzer untuk indikator masing-masing perintah.



Gambar 3.8 Konfigurasi Pixhawk dengan semua perangkat quadcopter

3.3 Perancangan Mekanik *Quadcopter*

Perancangan mekanik meliputi perhitungan beban *quadcopter*, desain awal *frame quadcopter* pada software *auto cad* dan *corel draw* hingga realisasi mekanik *quadcopter*.

3.3.1 Perhitungan Beban *Quadcopter*

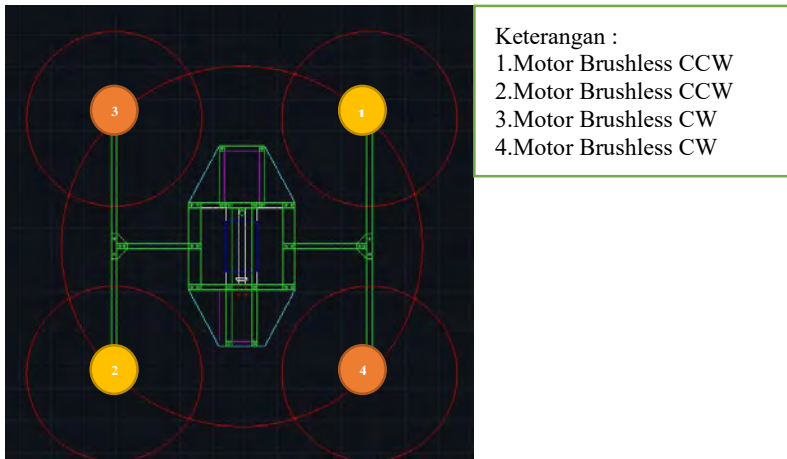
Dalam tugas akhir ini, tabel 3.1 menunjukkan estimasi beban yang akan dibawa *quadcopter*.

Tabel 3.1 Estimasi Berat Quadcopter

No	Nama Komponen	Jumlah	Berat (gram)	Berat Total
1	ESC 30A	4	40	160
2	Propeller	4	15	60
3	Motor	4	122	488
4	STM32F4	1	82	82
5	Raspberry Pi 2 + Modem	1	98	98
6	Pixhawk Kit	1	220	220
7	Sensor MQ-7	1	135	135
8	Frame Aluminium	1	500	500
9	Baterai Lipo 4 Sel	1	500	500
10	Baterai Lipo 3 Sel	1	78	78
11	Power Bank	1	110	110
12	Lampu	4	15	60
TOTAL BERAT				2491

3.3.2 Desain Awal *Quadcopter*

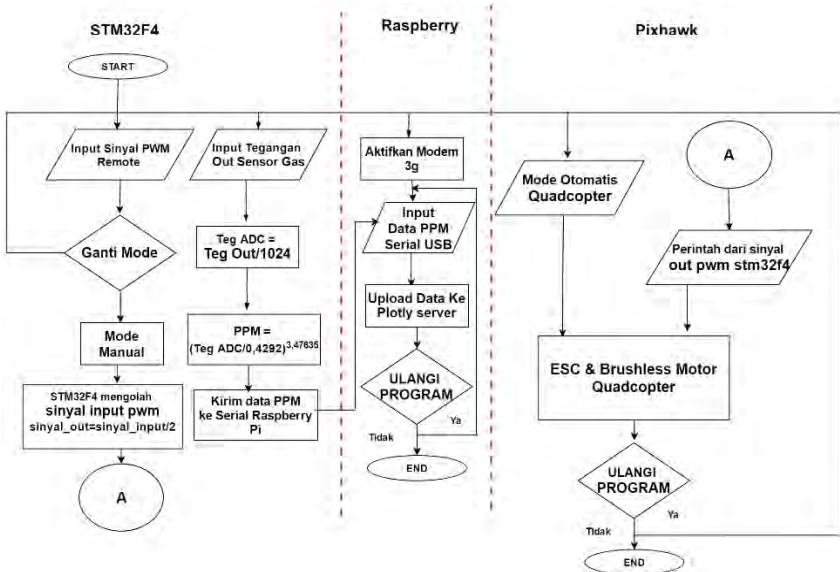
Pada subbab 3.3.1 telah diperhitungkan estimasi beban yang akan dibawa oleh *quadcopter*. Dalam desain awal *quadcopter*, rangka yang akan diterapgunakan menggunakan ukuran s700 (diagonal antar motor 1 dan 3 sepanjang 70cm). Terdapat 4 *brushless motor* DC yang dipasang pada rangka sesiao dengan gambar 3.9. Perancangan ukuran total *quadcopter* pada tugas akhir ini (panjang * lebar) 55x46 cm.



Gambar 3.9 Desain Quadcopter pada Auto Cad

3.4 Perangkat lunak

Perangkat lunak yang dirancang terdiri atas tiga bagian seperti pada gambar 3.10 yaitu perangkat lunak pada STM32F4, perangkat lunak pada pixhawk menggunakan *mission planner*, dan perangkat lunak pada raspberry pi 2. Perangkat lunak pada STM32F4 berupa perangkat lunak untuk melakukan proses ADC sebagai konversi data sensor analog menjadi data PPM dan pemrosesan input berupa sinyal PWM dari *remote control*. Untuk *mission planner*, merupakan proses deklarasi titik *waypoint* dan kontrol *quadcopter* di pixhawk. Sedangkan raspberry pi 2 berguna sebagai web server berbasis *web socket* sehingga dapat menampilkan grafik data ppm secara *online real time*.



Gambar 3.10 Blok sistem perangkat lunak *quadcopter*

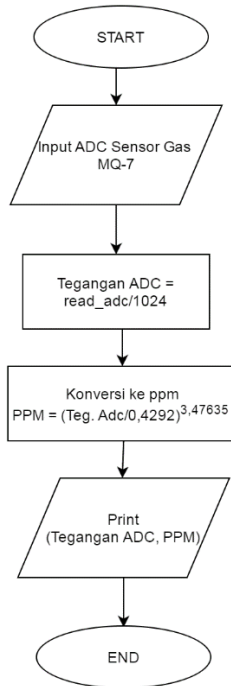
3.4.1 Perangkat Lunak STM32F4

Perangkat lunak STM32 secara umum digunakan untuk mengkonversikan data *analog* menjadi data *digital* dengan memanfaatkan ADC internal. Proses pengolahan sinyal PWM *remote control* menggunakan pin timer. Proses ini menggunakan *software coocox* dalam STM32F4 menggunakan clock 12 MHz, *USART transmitter* dengan *Baud Rate* 9600, pin timer 1, 3, 4, 8, dan ADC yang digunakan 10 bit.

3.4.1.1 Perangkat Lunak ADC STM32F4

Dalam tugas akhir ini menggunakan ADC dengan fidelitas 10 bit dan alur pembacaan ADC sesuai dengan gambar 3.11. Perhitungan tegangan ADC (*Analog Digital Converter*) dapat dirumuskan pada persamaan 3.1 :

$$\text{Tegangan ADC} = \frac{V_{in} \times 1024}{V_{ref}} \quad (3.1)$$



Gambar 3.11 Flowchart Program ADC STM32F4

Dan dalam aplikasi program pembacaan data sensor adalah sebagai berikut:

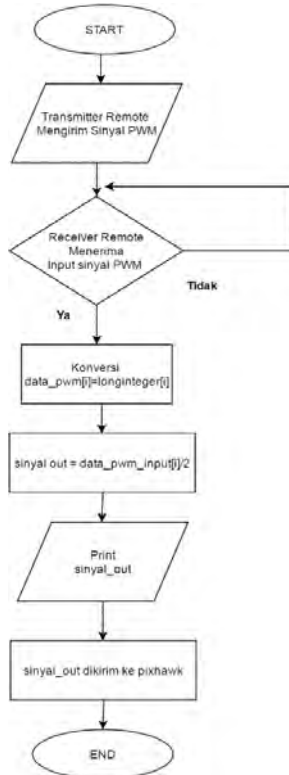
```

for(int x=0; x<100;x++)
{
    sensor_value1 = sensor_value1 + hasil_konversi_adc[0];
}
sensor_value1 = sensor_value1/100.0;

sensor_volt1 = sensor_value1/1024*4.9;
  
```

Dari program diatas data tersebut port yang digunakan adalah Port A0. Lalu program dikonversikan menggunakan algoritma dari hasil kalibrasi sensor gas MQ-7 pada gas karbon monoksida yang telah diketahui.

3.4.1.2 Perangkat Lunak Remote Control STM32F4



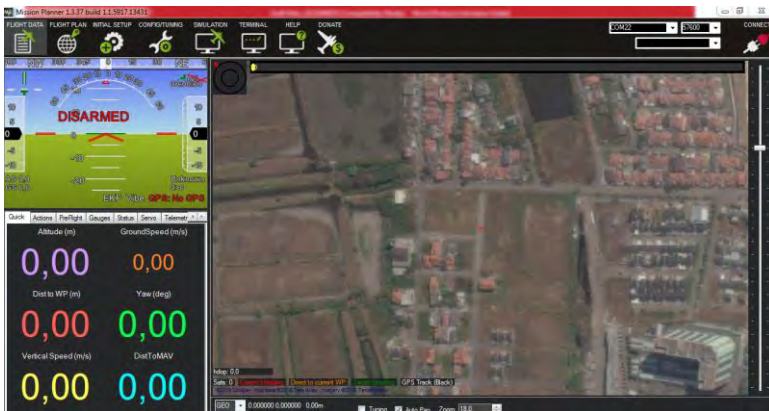
Gambar 3.12 Flowchart Program *Remote Control*

Pada tugas akhir ini digunakan *Remote control* radiolink AT-9 sebagai pengontrol *quadcopter*. Supaya data sinyal PWM *remote control* dapat dibaca oleh pixhawk, maka digunakan pin timer oleh STM32F4 untuk mengolah data sinyal PWM *remote control*.

Sesuai gambar 3.12 program *remote control* dibagi menjadi 2, yaitu saat mengolah input sinyal PWM remote dan mengeluarkan output sinyal PWM hasil pengolahan di STM32F4. Berikut merupakan untuk olah input sinyal PWM pada timer input (pin timer 1 dan 8), sehingga menghasilkan sinyal output PWM pada timer output (pin timer 3 dan 4).

3.4.2 Perangkat Lunak *Mission Planner*

Mission planner merupakan software untuk menunjang controller *quadcopter*, UAV, dan drone berupa *ardupilot* atau *pixhawk*. Pada tugas akhir ini, *mission planner* berperan sebagai penunjang controller *pixhawk*. Deklarasi titik-titik *waypoint* untuk melakukan misi terbang secara otomatis, pemetaan, dapat dilakukan melalui software ini. Kalibrasi sensor-sensor IMU, GPS, dan kompas pada *pixhawk* juga dilakukan melalui *mission planner*.



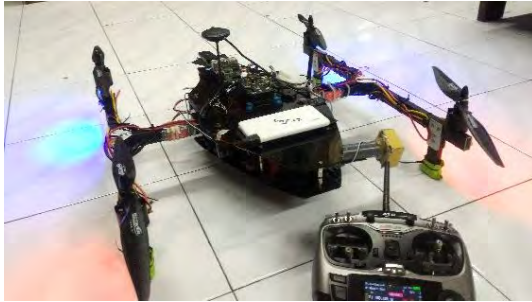
Gambar 3.13 Tampilan awal software mission planner

Pada gambar 3.13, untuk deklarasi misi dapat dilakukan pada tab *flight plan*, lalu pada bagian perintah *waypoint* di kolom bawah pilih “*add below*”. Maka titik-titik *waypoint* dapat dideklarasikan, dengan cara mengklik ke gambar peta yang tertera. Dalam menjalankan misi, *quadcopter* harus mendapatkan nilai GPS yang sesuai pada posisi *waypoint* yang akan dilaksanakan. Jika tidak, maka *quadcopter* akan *fly away* atau terbang dengan sendirinya tanpa perintah. Setelah titik-titik *waypoint* dideklarasikan, program dimasukkan ke *pixhawk* melalui *telemetry* 433 MHz dengan menekan tombol *Write*.

.....*Halaman ini sengaja dikosongkan*.....

BAB IV

PENGUJIAN DAN ANALISIS SISTEM

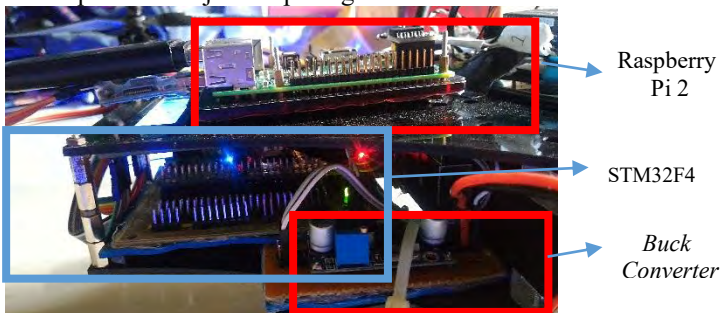


Gambar 4.1 *Quadcopter* yang diterapkan

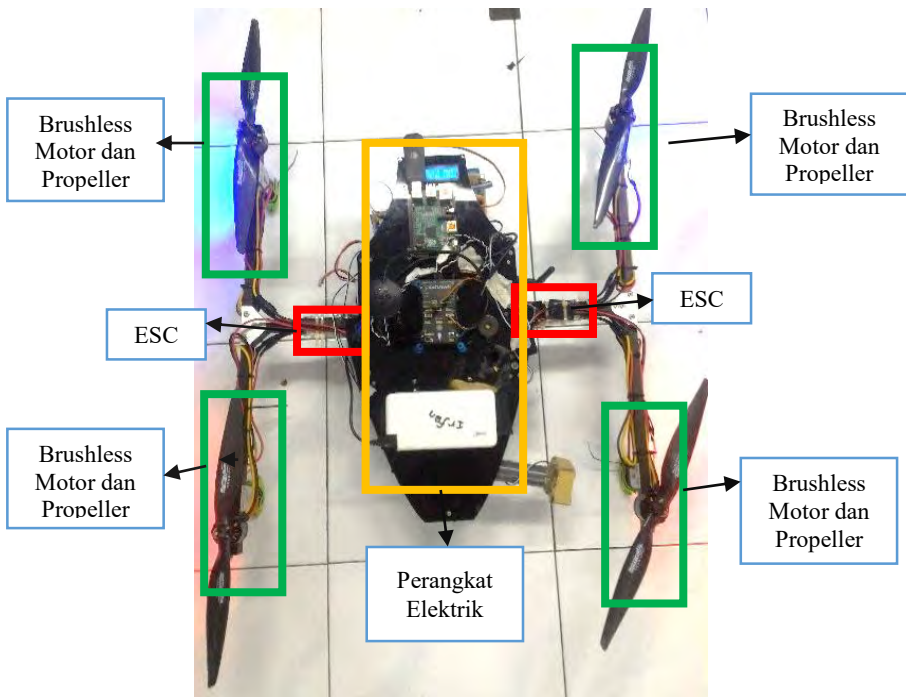
Pengujian sistem *quadcopter* dibagi menjadi beberapa tahap bagian, dimulai dari pengujian perangkat keras dan perangkat lunak. Quadcopter yang digunakan sesuai pada gambar 4.1. Tujuan dari pengujian dari berbagai aspek dari perancangan ini untuk mendapatkan parameter atau evaluasi performa dari *quadcopter*.

4.1 Realisasi Desain *Quadcopter*

Rangka *quadcopter* menggunakan aluminium sesuai dengan ukuran yang telah didesain pada software *auto cad*. *Quadcopter* memiliki ukuran yang sama dengan desain, yaitu 55x46 cm. Realisasi peletakkan komponen-komponen ditunjukkan pada gambar 4.2a dan 4.2b.



(a)



(b)

Gambar 4.2 Realisasi desain quadcopter yang telah dirakit, (a) tampak samping quadcopter, (b) tampak atas quadcopter

4.2 Pengujian Perangkat Keras

Pengujian perangkat keras dilakukan dengan menguji hasil output dari tiap sub rangkaian elektrik secara keseluruhan. Pengujian yang dilakukan meliputi pengujian dari *supply* dari *buck converter* yang digunakan, pengujian kalibrasi sensor, pengujian *remote control*, dan pengujian GPS untuk *waypoint*. Pengujian dilakukan agar memenuhi batasan – batasan masalah pada tugas akhir ini.

4.2.1 Pengujian *Supply Buck Converter*

Rangkaian *power supply* merupakan rangkaian terpenting yang digunakan sebagai sumber daya dari seluruh sub rangkaian elektrik. Perangkat *supply* berfungsi mengkonversikan tegangan DC dari baterai lipo 3 sel menjadi tegangan DC 5 volt. Rangkaian ini menjadi satu dengan suplai 12 volt untuk menggerakkan kipas sensor gas.



(a)



(b)

Gambar 4.3 (a) Pengujian V_{in} 12 volt, dan (b) Pengujian V_{out} 5 volt

Untuk mendapatkan tegangan 5V pada rangkaian *buck converter*, diperlukan memutar *multitune* hingga mendapatkan nilai output tegangan seperti gambar 4.3b. Tegangan 13,01 volt tersebut diinputkan ke kipas sensor gas, sedangkan output 5 volt digunakan input tegangan STM32F4 dan sensor gas MQ-7.

4.2.2 Pengujian Kalibrasi Sensor

Rangkaian sensor merupakan rangkaian pembagi tegangan sensor MQ-7. Pada pengujian ini tegangan masukan pada Rangkaian sensor 5 Volt. Hal yang di uji dari Rangkaian sensor adalah V_{in} , dan tegangan output pada sensor. Pengujian ini bertujuan untuk mendapatkan beberapa nilai, yaitu :

1. Nilai tegangan output saat diberi kadar gas CO yang berbeda.
2. Rumus algoritma dari nilai tegangan output dengan kadar PPM dari CO-meter untuk mengkonversi ke dalam bentuk *part per million* (PPM).

4.2.2.1 Prosedur Kalibrasi Awal Sensor Gas MQ-7

Pengujian kalibrasi sensor gas CO menggunakan alat *Carbon Monoxide* meter (CO-meter) model 7701 *az instrument* yang dapat

mengukur kadar gas CO mulai dari 0-999 ppm di udara. Dengan tingkat akurasi $\pm 20\%$ saat kadar gas CO 0-100 ppm dan $\pm 15\%$ saat kadar gas CO 100-500 ppm.



AZ-0016

Gambar 4.4 Alat CO-meter model 7701 *Az Instrument*

Supaya CO-meter dapat digunakan untuk mengukur gas karbon monoksida, maka diperlukan beberapa prosedur kalibrasi awal seperti gambar 4.5 :



Gambar 4.5 Tampilan awal CO-meter model 7701 *Az Instrument*

1. Saat Co-meter dinyalakan pertama, maka LCD akan berkedip-kedip seperti gambar 4.5. CO-meter melakukan kalibrasi awal dengan udara bersih dahulu.



Gambar 4.6 Tampilan CO-meter setelah kalibrasi dengan udara bersih

2. Setelah beberapa detik, CO-meter terkalibrasi dan LCD akan menunjukkan seperti gambar 4.6 dimana terdapat tulisan 0 ppm. CO-meter dapat digunakan untuk mendeteksi gas CO.
3. Untuk kalibrasi sensor gas MQ-7, maka CO-meter diletakkan pada sebuah wadah tertutup yang telah diberi gas CO hasil dari pembakaran kardus. Hal ini dilakukan agar gas CO tidak tercampur dengan udara lain.



Gambar 4.7 Pengujian kalibrasi sensor menggunakan CO-meter

4. Tegangan output yang terukur di avometer terdapat selisih 0,03 V – 0,05 V dengan hasil pembacaan di tegangan ADC arduino mega. Tabel 4.1 menunjukkan hasil pembacaan tegangan output sensor pada multimeter dan dari serial monitor arduino uno.



(b)

```
sensor_volt = 1.29V
RS = 28.91
rat = 1.44
```

(b)

Gambar 4.8 (a) Tegangan output sensor di multimeter, dan (b) Tegangan ADC output sensor di serial monitor Arduino

Tabel 4.1 Perbedaan tegangan multimeter dengan tegangan ADC

No	Tegangan Multimeter	Tegangan ADC
1	0,47 volt	0,51 volt
1	1,1 volt	1,4 volt
2	1,22 volt	1,24 volt
3	1,35 volt	1,39 volt
4	1,44 volt	1,47 volt
5	1,61 volt	1,65 volt
6	1,91 volt	1,94 volt

4.2.2.2 Pencarian Model Matematis Kalibrasi Sensor Gas MQ-7

Pada tugas akhir ini, alat bertujuan untuk mendeteksi kadar polutan gas karbon monoksida (CO) dalam bentuk satuan PPM. Agar mendapatkan konversi satuan PPM, maka diperlukan mencari fungsi algoritma sensor saat diberi polutan gas CO. Dengan nilai tegangan output yang telah diketahui, maka hal yang dilakukan saat pengujian dibandingkan dengan kadar PPM pada CO-meter. Berikut hasil kalibrasi sensor gas ditunjukkan pada tabel 4.2 :

Tabel 4.2 Hasil Tegangan Output Sensor Gas MQ-7 dengan PPM

No	Tegangan Out	PPM
1	0,59	3
2	0,77	10
3	1,058	21
4	1,17	32
5	1,268	40
6	1,391	53
7	1,476	65
8	1,524	73
9	1,604	85
10	1,644	100
11	1,69	114
12	1,703	136
13	1,717	151
14	1,748	165
15	2,073	211
16	2,12	264
17	2,21	322
18	2,32	350
19	2,41	383

Dari tabel 4.2 didapatkan hasil tegangan output dengan kadar gas CO dalam satuan ppm yang diukur dengan CO-meter. Selanjutnya hubungan antara tegangan output dengan kadar gas ppm dituangkan dalam bentuk grafik kemudian dicari model matematisnya.

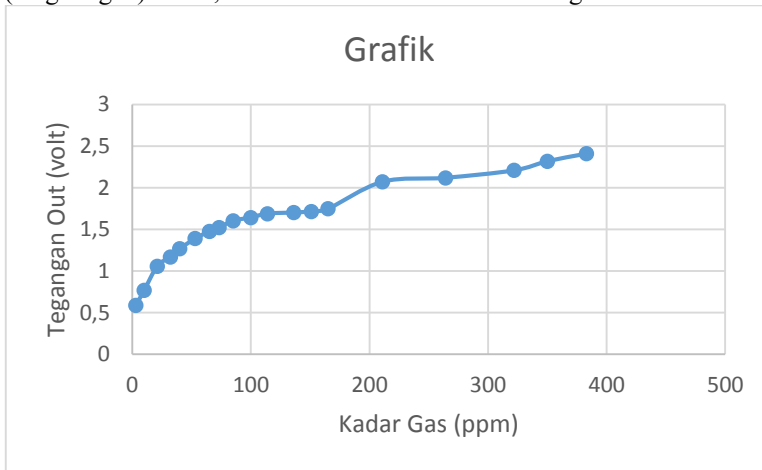
Untuk mencari model matematis suatu persamaan garis atau kurva menggunakan metode regresi (*trendline*) dalam *microsoft excel*. Regresi linier digunakan untuk menentukan fungsi linier yang paling sesuai dengan kumpulan titik data (x,y) yang diketahui. Regresi linier memiliki batasan, yaitu hanya digunakan untuk kurva yang berbentuk mendekati garis lurus. Berikut merupakan persamaan yang digunakan untuk regresi linier :

$$\text{Linear} : Y = a + b X \quad (4.1)$$

$$\text{Multi linear} : Z = a + b X + c Y \quad (4.2)$$

Dimana Y = data di sumbu Y, a = konstanta, b = gradien, X = data di sumbu X, Z = hasil regresi linear.

Tabel 4.2 bila diubah ke model grafik, maka didapatkan *slopes* (lengkungan) kurva, dimana tidak mendekati bentuk garis lurus.



Gambar 4.9 Hasil data tabel 4.2 dalam bentuk grafik *excel*

Gambar 4.9 menunjukkan bahwa setiap kenaikan nilai X maka data nilai Y akan turut naik namun tidak linier. Kurva yang didapatkan memiliki pola (*trend*) lengkung. Agar mendapatkan model matematis dari grafik tersebut, perlu dilakukan operasi koordinat agar kurva lengkung tersebut dapat direpresentasikan dalam kurva liner. Sehingga digunakan analisis regresi dengan menggunakan fungsi *power*. Linearisasi kurva lengkung tersebut dapat menggunakan fungsi *power* seperti persamaan 4.3 :

$$Y = aX^b \quad (4.3)$$

$$\text{Log}(Y) = \text{log}(a X^b)$$

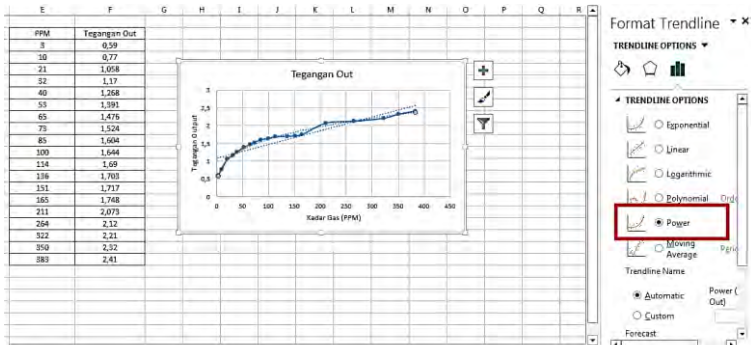
$$\text{Log}(Y) = \text{log}(a) + b \text{log}(X)$$

Jika : $P = \text{log}(Y)$; $A = \text{log}(a)$; $B = b$; $Q = \text{log}(X)$;
maka :

$$P = A + Bq \quad (4.4)$$

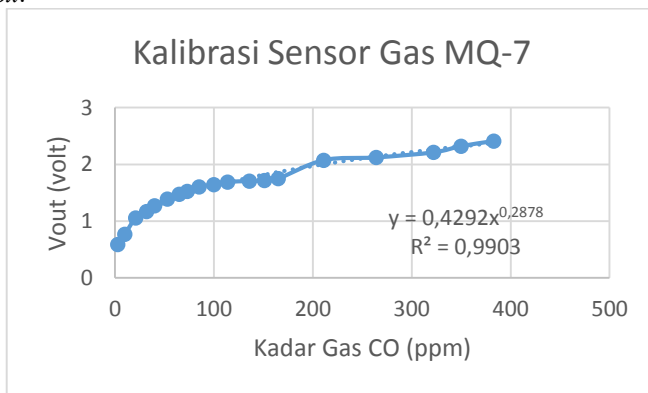
Dimana Y = data sumbu Y, a = konstanta, X = data sumbu X, b = konstanta pangkat.

Setelah diketahui menggunakan metode regresi (*trendline*) yang sesuai dengan data tabel 4.2, maka dicari model matematis menggunakan *trendline power* di *microsoft excel* seperti gambar 4.10.



Gambar 4.10 Hasil kalibrasi kurva regresi linier menggunakan *trendline* power

Didapatkan fungsi regresi (*trendline*) linierisasi kurva lengkung sesuai dengan hasil input tegangan output dan kadar gas. Pada gambar 4.11 didapatkan fungsi $y = 0,4292x^{-0,2878}$ dan akurasi antar data (R^2) sebesar 99,03 %. Untuk memasukkan fungsi ke mikrokontroller STM32F4, data yang ingin dicari adalah data kadar gas ppm (sumbu X) dengan data tegangan output (sumbu Y) yang diketahui. Sehingga fungsi dibalik menjadi $x = (y/0,4292)^{3,47635}$ dan dilakukan komputasi pada *software coocox*.



Gambar 4.11 Fungsi algoritma dari kalibrasi sensor gas MQ-7

4.2.2.3 Perbandingan Nilai Error Regresi Power dengan Regresi Lainnya

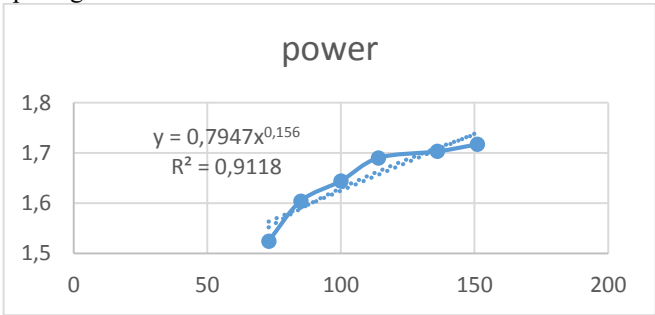
Terdapat beberapa jenis regresi yang ada di *excel*. Untuk mengetahui apakah model matematis tersebut bekerja secara efisien pada sensor gas, maka perlu diketahui perbandingan nilai error. Data-data tersebut diambil 5 sampel (tabel 4.3) untuk dibandingkan dengan beberapa metode regresi (*trendline*) yang ada di *excel*. Dari beberapa metode regresi hasilnya dibandingkan dengan kadar gas ppm di CO-meter. Berikut persamaan 4.5 untuk mengukur error :

$$Error = \frac{[Hasil\ pengukuran - Hasil\ rumus\ regresi]}{Hasil\ pengukuran} \times 100\% \tag{4.5}$$

Tabel 4.3 Sampel data yang digunakan perhitungan error

Hasil Pengukuran (ppm)	Tegangan Out (volt)
73	1,524
85	1,604
100	1,644
114	1,69
136	1,703
151	1,717

Error yang didapatkan saat menggunakan regresi (*trendline*) *power* digunakan sebagai referensi perbandingan dengan metode regresi lainnya. Didapatkan error rata-rata regresi power sebesar 7,25% dengan fungsi pada gambar 4.12.



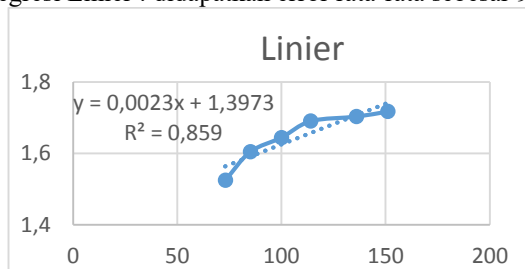
Gambar 4.12 Grafik fungsi algoritma dari sampel regresi power

Tabel 4.4 Perhitungan error regresi power

Hasil Pengukuran (ppm)	Tegangan Out (volt)	Hasil ppm Rumus Power	Error
73	1,524	64,97	11,02 %
85	1,604	90,18	6,1 %
100	1,644	105,61	5,61 %
114	1,69	126,04	10,56 %
136	1,703	132,39	2,65 %
151	1,717	139,52	7,59 %
Error Rata-Rata			7,25 %

Perbandingan dilakukan dengan 3 metode regresi lainnya, yaitu : regresi linier, logaritmik, dan eksponensial menggunakan *microsoft excel*.

1. Regresi Linier : didapatkan error rata-rata sebesar 9,89%.

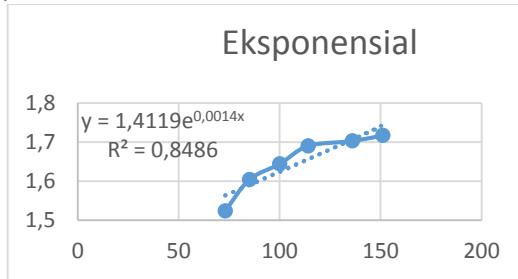


Gambar 4.13 Grafik fungsi algoritma dari sampel regresi linier

Tabel 4.5 Perhitungan error regresi linier

Hasil Pengukuran (ppm)	Tegangan Out (volt)	Hasil ppm Rumus Linier	Error
73	1,524	55,08	24,53 %
85	1,604	89,87	5,72 %
100	1,644	107,26	7,26 %
114	1,69	127,26	11,63 %
136	1,703	132,91	2,26 %
151	1,717	139	7,94 %
Error Rata-Rata			9,89 %

2. Regresi Eksponensial : didapatkan error rata-rata sebesar 10,46%.

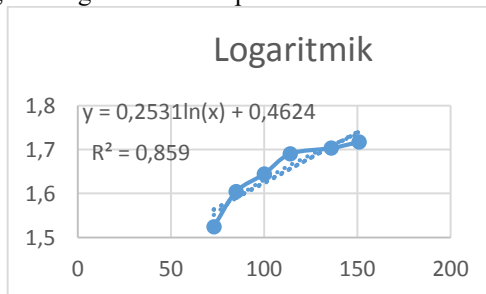


Gambar 4.14 Grafik fungsi algoritma dari sampel regresi eksponensial

Tabel 4.6 Perhitungan error regresi eksponensial

Hasil Pengukuran (ppm)	Tegangan Out (volt)	Hasil ppm Rumus Linier	Error
73	1,524	54,57	25,24 %
85	1,604	91,11	7,19 %
100	1,644	108,71	8,71 %
114	1,69	128,42	12,65 %
136	1,703	133,89	1,54 %
151	1,717	139,74	7,45 %
Error Rata-Rata			10,46 %

3. Regresi Logaritmik : didapatkan error rata-rata sebesar 8,24%



Gambar 4.15 Grafik fungsi algoritma dari sampel regresi logaritmik

Tabel 4.7 Perhitungan error regresi logaritmik

Hasil Pengukuran (ppm)	Tegangan Out (volt)	Hasil ppm Rumus Linier	Error
73	1,524	68,68	5,91 %
85	1,604	94,46	11,13 %
100	1,644	110,78	10,78 %
114	1,69	133,06	16,72 %
136	1,703	140,13	3,04 %
151	1,717	148,17	1,86 %
Error Rata-Rata			8,24 %

Kesimpulan :

Dari keempat metode regresi yang diuji, metode regresi power memiliki error terkecil sebesar 7,25%. Regresi power mampu membuat kurva lengkung terlinierisasi. Sehingga regresi ini dapat diterapkan untuk mencari model matematis dari kalibrasi sensor gas MQ-7.

4.2.2.4 Perbandingan Nilai Sensor Gas MQ-7 dengan CO-meter

Untuk mengetahui error, diambil beberapa sampel dari data ppm sensor gas yang dibandingkan dengan data CO-meter menggunakan persamaan 4.6 :

$$Error = \frac{|Hasil\ COmeter - Hasil\ Sensor\ Gas|}{Hasil\ COmeter} \times 100\% \quad (4.6)$$

Fungsi hasil kalibrasi dimasukkan ke dalam program STM32F4 dan ditampilkan pada sebuah LCD 16x2. Pengujian dilakukan dengan membakar kardus sehingga menimbulkan polutan. Pembakaran diletakkan pada sebuah wadah tertutup agar tidak terkena interferensi dari gas-gas lain.

Dari hasil pengujian kalibrasi sensor gas MQ-7 dapat mendeteksi gas karbon monoksida dan dikonversi ke dalam satuan *part per million* (PPM). Pada tabel 4.8 menunjukkan bahwa terdapat error pembacaan $\pm 9,4\%$ (2-5 PPM) dibandingkan dengan CO-meter. Hal ini bisa terjadi karena adanya perubahan tegangan input dari sensor gas MQ-7 dan nilai sensitifitas sensor gas MQ-7 yang lebih rendah dari CO-meter.

Tabel 4.8 Perbandingan nilai sensor gas mq-7 dengan CO-meter

No	CO-meter (ppm)	Sensor Gas (ppm)	Error (%)
1	3	3,02	0,67
2	10	7,62	23,8
3	12	12,27	2,25
4	14	11,97	14,5
5	21	22,98	9,43
6	22	23,35	6,13
7	24	26,71	11,29
8	32	32,6	1,875
9	53	59,48	12,22
10	56	62,66	11,89
Error Rata-Rata			9,4

4.2.3 Pengujian *Remote Control*



Gambar 4.16 Pengujian nilai input sinyal PWM remote pada STM32F4

Remote control yang digunakan dalam pengujian adalah radiolink at-9 yang memiliki 9 channel sinyal. Pada tugas akhir ini, hanya 6 channel saja yang dipakai untuk mengatur gerakan dan mode terbang dari *quadcopter*. Input sinyal PWM dari *remote control* dimasukkan ke pin

timer 1 dan 8 STM32F4. Tabel 4.9 menunjukkan nilai sinyal PWM masing-masing channel sinyal yang diinputkan ke STM32F4 dan ditampilkan pada LCD 16x2 seperti pada gambar 4.16.

Tabel 4.9 Nilai Sinyal PWM masing-masing channel

No	Channel	Nilai Sinyal PWM		
		Low	Medium	High
1	Channel 1	2177	3006	3827
2	Channel 2	2176	3008	3838
3	Channel 3	2176	3162	3837
4	Channel 4	2177	3014	3830
5	Channel 5	2173	3009	3842
6	Channel 6	2174	-	3843

Dari hasil pengujian didapatkan rentang sinyal PWM yang terbaca pada STM32F4 antara 2000-4000. Nilai sinyal PWM yang dapat dibaca oleh pixhawk adalah antara 1000-2000. Agar sinyal PWM dapat dibaca pixhawk, maka pada program STM32F4 diberi algoritma untuk mengompensasi nilai sinyal PWM dengan cara membagi 2 nilai sinyal. Berikut algoritma program *coocox* untuk pengolahan sinyal PWM :
Definisi awal pembacaan sinyal PWM *remote control* :

```
signed long int lebar_pulsa1;
signed long int lebar_pulsa2;
signed long int lebar_pulsa3;
signed long int lebar_pulsa4;
signed long int lebar_pulsa5;
signed long int lebar_pulsa6;
signed long int lebar_pulsa7;
signed long int lebar_pulsa8;
```

Fungsi kompensasi input sinyal PWM :

```
void olah_pwm_remot()
{
    out1=lebar_pulsa1/2;
    out2=lebar_pulsa2/2;
    out3=lebar_pulsa3/2;
    out4=lebar_pulsa4/2;
    out5=lebar_pulsa5/2;
    out6=lebar_pulsa6/2;
}
void channel_6_remot()
```



```

{
    baca_kondisi_channel_6();
    switch(state_channel_6)
    {
        case 0:                                     //full manual
            olah_pwm_remot();

            //baca_kondisi_channel_5();
            counter1=0;
            break;

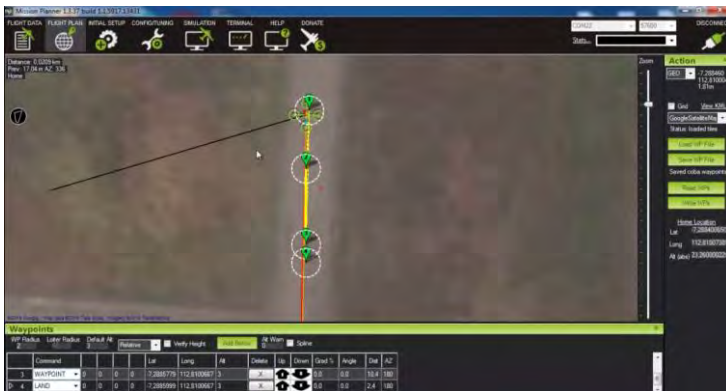
        case 1:                                     //misi
            olah_pwm_remot2();
            channe_5_auto();
            break;

        case 2:                                     //reset
            olah_pwm_remot();
            state_channel_5=1;
            counter1=0;
            break;
    }

    state_channel_6_sebelum=state_channel_6;
}
}

```

4.2.4 Pengujian GPS dan Waypoint *Quadcopter*



Gambar 4.17 Perintah waypoint yang dideklarasikan pada *mission planner*

Pada tugas akhir ini, *quadcopter* bertugas melaksanakan misi sesuai gambar 4.17 dengan titik-titik *waypoint* yang dideklarasikan pada *mission planner*. Supaya dapat berjalan dengan sempurna, maka GPS dan kompas terlebih dahulu dikalibrasi. Saat *quadcopter* pertama kali

dinyalakan, maka pixhawk akan menunggu beberapa saat untuk *lock* nilai GPS di tempat tersebut. Apabila pixhawk berhasil mendapatkan sinyal GPS, maka saat *take off* pertama, posisi tersebut dijadikan posisi *home*.

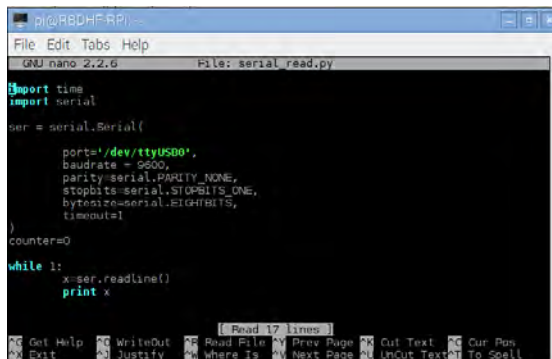
Tabel 4.10 Nilai GPS Pengujian *Waypoint*

No	Jenis Misi	Altitude	Latitude	Longitude
1	<i>Home Location</i>	23,26	-7,28840650	112,8100738
2	<i>Waypoint 1</i>	3	-7,288412	112,810071
3	<i>Waypoint 2</i>	3	-7,288484	112,810067
4	<i>Waypoint 3</i>	3	-7,288578	112,810067
5	<i>Land</i>	0	-7,2886	112,810067

Pada hasil pengujian ini didapatkan nilai GPS yang fix dan *quadcopter* dapat melalui semua titik *waypoint* seperti pada tabel 4.10.

4.3 Perangkat Lunak Raspberry Pi 2

Raspberry pi 2 berguna sebagai *data logger* dan menerima data kadar gas dari STM32F4 melalui USB to TTL. Data yang diterima berupa data serial *float*. Program yang digunakan sesuai gambar 4.18



```

pi@RBDHF-PI:~$
File Edit Tabs Help
GNU nano 2.2.6 File: serial_read.py

import time
import serial

ser = serial.Serial(
    port='/dev/ttyUSB0',
    baudrate = 9600,
    parity=serial.PARITY_NONE,
    stopbits=serial.STOPBITS_ONE,
    bytesize=serial.EIGHTBITS,
    timeout=1
)
counter=0

while 1:
    x=ser.readline()
    print x
  
```

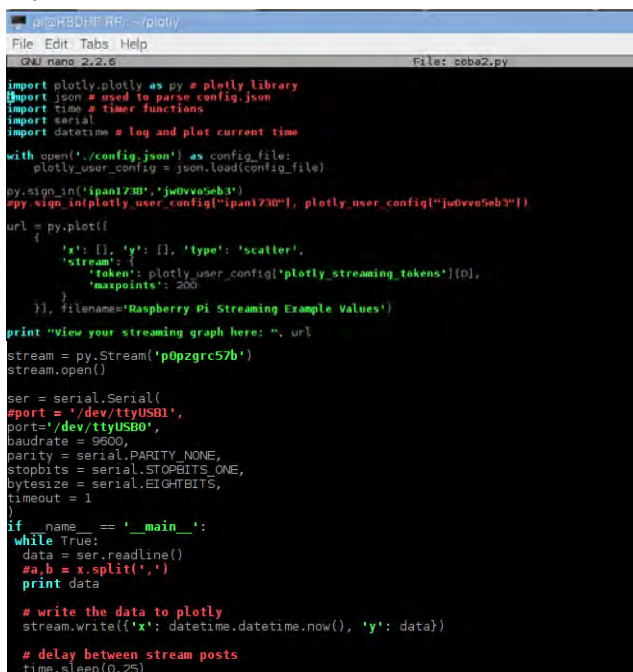
Gambar 4.18 Program pembacaan data serial USB

Agar data dapat dilihat secara online dan *real time*, maka raspberry pi 2 menggunakan bahasa pemrograman python dan aplikasi *plotly*. Pembacaan data serial yang dikirim oleh STM32F4, dibaca oleh raspberry

pi 2 melalui port USB0. Program menggunakan bahasa pemrograman python yang diakses melalui terminal raspberry pi.

4.3.1 Program Plotly Raspberry Pi 2

Plotly merupakan sebuah website yang memfasilitasi pengguna untuk memplotting data dari raspberry pi, mikrokontroler, data log industri, dan sebagainya. Website ini menyajikan fitur database berbasis *web socket*, sehingga pengguna diwajibkan memasukkan *username*, *API Keys*, dan *Streams Key*. Pada tugas akhir ini, kadar gas berupa ppm ditampilkan secara *real time* dan diplot menggunakan grafik *scatter*. Agar data serial dari USB dapat ditampilkan, maka diperlukan tambahan program python untuk mengakses *plotly*. Berikut list program pada gambar 4.19 :



```
pi@RSDPi:~$ python3 plotly.py
File Edit Tabs Help
GNU nano 2.2.6 File: sobaz2.py

import plotly.plotly as py # plotly library
import json # used to parse config.json
import time # timer functions
import serial
import datetime # log and plot current time

with open('./config.json') as config_file:
    plotly_user_config = json.load(config_file)

py.sign_in('ipani1730','ju0vvo5eb3')
py.sign_in(plotly_user_config["ipani1730"], plotly_user_config["ju0vvo5eb3"])

url = py.plot(
    {
        'x': [], 'y': [], 'type': 'scatter',
        'stream': {
            'token': plotly_user_config["plotly_streaming_tokens"][0],
            'maxpoints': 200
        }
    }, filename='Raspberry Pi Streaming Example Values')

print "View your streaming graph here: ", url

stream = py.Stream('p0pzgrc57b')
stream.open()

ser = serial.Serial(
    #port = '/dev/ttyUSB1',
    port='/dev/ttyUSB0',
    baudrate = 9600,
    parity = serial.PARITY_NONE,
    stopbits = serial.STOPBITS_ONE,
    bytesize = serial.EIGHTBITS,
    timeout = 1
)

if __name__ == '__main__':
    while True:
        data = ser.readline()
        #a,b = x.split(',')
        print data

        # write the data to plotly
        stream.write({'x': datetime.datetime.now(), 'y': data})

        # delay between stream posts
        time.sleep(0.25)
```

Gambar 4.19 Program plotly raspberry pi

Untuk hasil dari program dapat diakses melalui akun yang telah dibuat sebelumnya. Hasil program dapat ditampilkan secara *public*

dengan mengganti setelan *privacy* menjadi *public*. Gambar 4.20 menunjukkan hasil dari pembacaan *plotly*.



Gambar 4.20 Hasil Pembacaan Kadar Gas PPM oleh STM32F4 di Plotly

4.4 Pengujian Seluruh Sistem

Pengujian seluruh sistem dilakukan dengan lengkap dimana keberhasilan pendeteksian gas dan *quadcopter* mampu melewati titik-titik *waypoint*. Pengujian dilakukan pada sebuah lapangan dengan ukuran 20x12 meter. Gas polutan berasal dari hasil pembakaran sampah pada titik-titik tertentu.

4.4.1 Pengujian Sistem Ke-1

Pengujian pertama dilakukan pada sebuah lapangan di daerah Sukolilo Park Regency dan *quadcopter* diposisikan seperti gambar 4.21. Pada pengujian berikut, nilai ketinggian (*altitude*) ditetapkan setinggi 3 meter. Pengujian dilakukan pada saat sampah dibakar di dua titik dalam jarak yang dapat dijangkau oleh titik *waypoint*. *Quadcopter* pertama digerakkan manual kemudian diubah mode ke auto menggunakan *channel 5 remote control radiolink*. Bila sinyal PWM *channel 5* bernilai di atas 1749, maka akan masuk mode otomatis.

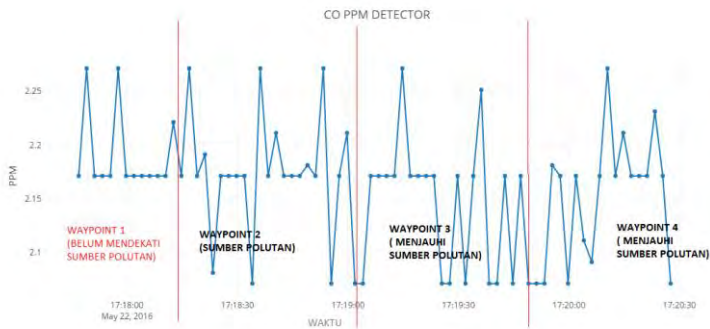


Gambar 4.21 Quadcopter saat melakukan awal misi pertama



Gambar 4.22 Lokasi pengujian sistem ke-1

Pembacaan sensor gas memanfaatkan aliran udara yang dihasilkan oleh hasil penyedotan dari propeller *quadcopter* lalu diteruskan menuju corong sensor gas. Titik-titik *waypoint* dideklarasikan sesuai dengan tabel 4.11. Pada pengujian ini, asap tidak dapat dihasilkan secara maksimal dikarenakan posisi angin yang kencang dan membuat asap langsung menghilang. Gambar 4.23 menunjukkan hasil data logger pada *plotly* :



Gambar 4.23 Hasil Data Logger pengujian pertama di Plotly

Tabel 4.11 Data kadar gas PPM saat pengujian pertama

No	Waktu	Kadar Gas PPM
1	2016-05-22 17:17:46.905610	2.17
2	2016-05-22 17:17:49.049322	2.27
3	2016-05-22 17:17:51.191512	2.17
4	2016-05-22 17:17:53.335121	2.17
5	2016-05-22 17:17:53.335121	2.27
6	2016-05-22 17:17:55.477525	2.17
7	2016-05-22 17:17:57.620046	2.17
8	2016-05-22 17:17:59.763630	2.17
9	2016-05-22 17:18:04.048619	2.17
10	2016-05-22 17:18:06.191979	2.22
11	2016-05-22 17:18:08.334236	2.27
12	2016-05-22 17:18:10.477987	2.27
13	2016-05-22 17:18:12.620473	2.17
14	2016-05-22 17:18:14.763004	2.17
15	2016-05-22 17:18:16.906769	2.19
16	2016-05-22 17:18:23.335356	2.08
17	2016-05-22 17:18:25.477727	2.17

Analisis :

Quadcopter mampu mengikuti sesuai dengan titik-titik *waypoint* yang dideklarasikan. Namun terdapat error jarak GPS sekitar 30-50 cm dari pembacaan GPS di *mission planner* dengan kenyataan. Hal ini masih dalam hal yang lumrah dikarenakan setelan *WP Radius* diset pada radius 2 meter. Sensor Gas terkena interferensi angin, sehingga data yang dihasilkan tidak sesuai dengan posisi polutan.

4.4.2 Pengujian Sistem Ke-2

Pengujian kedua dilakukan pada sebuah lapangan di daerah Sukolilo Park Regency dengan ukuran 25x12 meter. Gas polutan berasal dari hasil pembakaran sampah pada titik-titik tertentu. Pada pengujian berikut, nilai ketinggian (*altitude*) ditetapkan setinggi 4 meter.

Pengujian dilakukan saat dan setelah sampah dibakar. Pembakaran sampah dilakukan pada jarak yang dijangkau oleh titik *waypoint* yang dideklarasikan. Pembacaan sensor gas memanfaatkan aliran udara yang dihasilkan oleh hasil penyedotan propeller *quadcopter* lalu diteruskan menuju sensor gas semikonduktor.



Gambar 4.24 Lokasi pengujian dan posisi *quadcopter* saat awal misi kedua

Pada tabel 4.12, didapatkan nilai dari *longitude* (busur), *latitude* (lintang), dan *altitude* saat *quadcopter* melakukan misi *waypoint*. Dari hasil pengujian, *quadcopter* berhasil mengikuti semua titik *waypoint*.



Gambar 4.25 Deklarasi *waypoint* kedua dan tampilan *google maps* di *mission planner*

Tabel 4.12 Nilai GPS Pengujian *Waypoint* Kedua

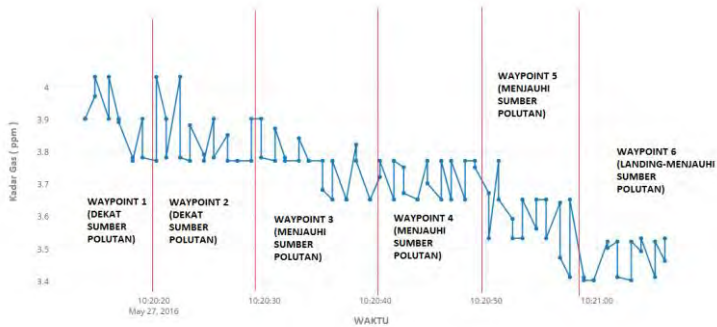
No	Jenis Misi	Altitude	Latitude	Longitude
1	Waypoint 1	4	-7,28843	112,810051
2	Loiter	4	-7,288515	112,810042
3	Waypoint 2	4	-7,288613	112,810035
4	Spline Waypoint	4	-7,288606	112,810085
5	Waypoint 3	4	-7,288551	112,810085
6	Loiter	4	-7,288499	112,810088
7	Waypoint 4	4	-7,28844	112,810093
8	Land	0	-7,288414	112,810069

Setelah pengujian terbang, maka data peta diolah menggunakan *software mission planner* dan *google earth*. Data kadar gas, diolah raspberry pi 2 menggunakan *plotly web server*. Gambar 4.26 menunjukkan hasil data logger pada *plotly*. Sedangkan, data posisi digambarkan dalam *3D analyzer google earth*, sehingga didapatkan hasil seperti gambar 4.28. Data PPM dan posisi *quadcopter* dipetakan pada

google earth. Tabel 4.13 menunjukkan hasil dari pembacaan sensor gas pada tempat yang dilalui *quadcopter*.

Tabel 4.13 Data kadar gas PPM pengujian kedua

No	Waktu	Kadar Gas (PPM)
1	10:20:13.809239	3.90
2	10:20:14.696126	3.97
3	10:20:15.951825	4.03
4	10:20:16.838881	3.90
5	10:20:21.125300	3.90
6	10:20:24.525468	3.79
7	10:20:26.668102	3.85
8	10:20:28.810378	3.77
9	10:20:30.953137	3.87
10	10:20:35.238035	3.68
11	10:20:36.125061	3.77
12	10:20:38.267464	3.82
13	10:20:43.811990	3.65
14	10:20:44.695927	3.77
15	10:20:51.125268	3.77
16	10:20:55.410632	3.65
17	10:20:58.811557	3.40
18	10:21:00.953557	3.52
19	10:21:06.125702	3.46



Gambar 4.26 Hasil Data Logger Pengujian Kedua di *Plotly*



(a)



(b)

Gambar 4.27 (a)Hasil 3D analyzer di *google earth* dan (b)hasil pemetaan kadar gas pengujian kedua.

Analisis :

Quadcopter mampu mengikuti sesuai dengan titik-titik *waypoint* yang dideklarasikan. Namun terdapat error jarak GPS pada *waypoint* 3 sebesar 40 cm dari pembacaan GPS di *mission planner* dengan kenyataan. Hal ini masih dalam hal yang lumrah dikarenakan setelan *WP Radius* diset pada radius 2 meter. Data peta dari 3D *analyzer* sesuai dengan hasil jalur yang dilewati oleh *quadcopter*.

4.4.3 Pengujian Sistem Ke-3

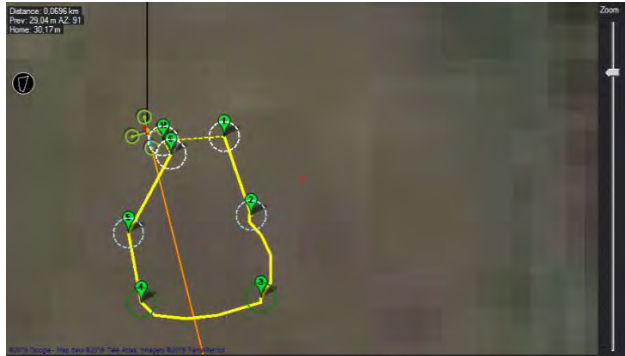
Pengujian kedua dilakukan pada sebuah lapangan di tempat parkir Teknik Industri dengan ukuran 30x15 meter. Gas polutan berasal dari hasil pembakaran sampah pada titik tertentu.



Gambar 4.28 Lokasi pengujian dan posisi *quadcopter* saat awal misi ketiga

Pada pengujian berikut, nilai ketinggian (*altitude*) ditetapkan setinggi 4 meter. Pengujian dilakukan saat dan setelah sampah dibakar. Pembakaran sampah dilakukan pada titik tertentu.

Pembacaan sensor gas memanfaatkan aliran udara yang dihasilkan oleh hasil penyedotan propeller *quadcopter* lalu diteruskan menuju sensor gas semikonduktor. Hasil pembacaan sensor gas MQ-7 di sekitar tempat pembakaran berkisar antara 2-5 ppm. Data kadar gas yang terukur dibandingkan dengan hasil pengukuran oleh *quadcopter*.



Gambar 4.29 Deklarasi *waypoint* ketiga dan tampilan *google maps* di *mission planner*

Pada tabel 4.14, didapatkan nilai dari *longitude* (busur), *latitude* (lintang), dan *altitude* saat *quadcopter* melakukan misi *waypoint*. Dari hasil pengujian, *quadcopter* berhasil mengikuti semua titik *waypoint*

Tabel 4.14 Nilai GPS Pengujian *Waypoint* Ketiga

No	Jenis Misi	Altitude	Latitude	Longitude
1	Waypoint 1	4	-7,284336	112,797474
2	Loiter (WP 2)	4	-7,2844315	112,7975187
3	Waypoint 3	4	-7,2845432	112,7975294
4	Waypoint 4	4	-7,2845512	112,7973658
5	Waypoint 5	4	-7,2844554	112,7973416
6	Loiter	4	-7,284357	112,79741

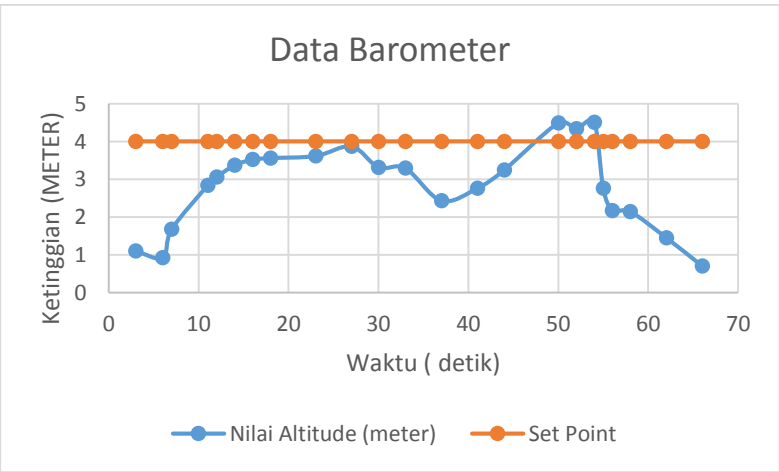
Saat pengujian didapatkan error ketinggian dari sensor barometer, dimana ketinggian yang nyata berbeda dengan *set point* awal. Perhitungan error mengambil beberapa sampel data barometer seperti pada tabel 4.15. Sesuai dengan persamaan 4.7, data barometer dikurangi data *set point* dan dibagi dengan data *set point* sehingga didapatkan nilai error ketinggian *quadcopter*.

$$Error = \frac{|Set\ Point\ Altitude - Data\ Barometer|}{Set\ Point\ Altitude} \times 100\% \quad (4.7)$$

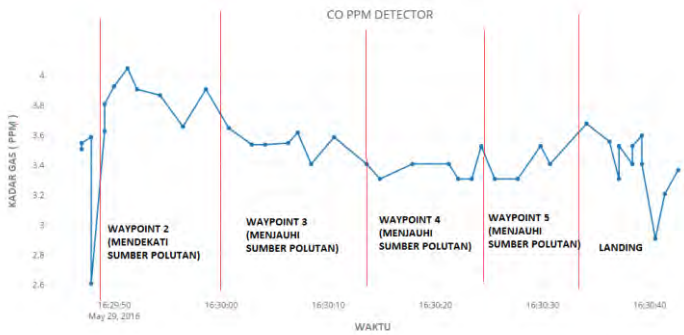
Tabel 4.15 Nilai sensor barometer saat pengujian ketiga

Waktu(Detik)	Nilai Altitude (meter)	Set Point	Error (%)
14	3,372125	4	15,69
16	3,525146	4	11,87
18	3,561984	4	10,95
23	3,615825	4	9,60
27	3,87086	4	3,22
37	2,4311331	4	39,22
41	2,762876	4	30,92
50	4,4913668	4	12,28
52	4,341257	4	8,53
55	2,760042	4	30,99
Error Rata- Rata			15,75

Data sensor barometer menunjukkan hasil data error ketinggian seperti pada tabel 4.15 dan gambar 4.30. *Set point* ketinggian pada 4 meter, namun hasil di kenyataan terdapat error sebesar 15,75% saat misi *waypoint*. Gambar 4.31 menunjukkan hasil data logger pada *plotly*.



Gambar 4.30 Error ketinggian *quadcopter* berdasarkan sensor Barometer



Gambar 4.31 Hasil Data Logger Pengujian Ketiga di *Plotly*

Tabel 4.16 Data kadar gas PPM

No	Waktu	Kadar Gas (PPM)
1	16:29:46.960723	3.5
2	16:29:46.960723	3.54
3	16:29:47.844330	3.58
4	16:29:51.244025	4.04
5	16:29:54.272578	3.86
6	16:29:56.416153	3.65
7	16:30:04.103290	3.53
8	16:30:06.245508	3.54
9	16:30:10.531799	3.58
10	16:30:13.559195	3.4
11	16:30:14.816906	3.3
12	16:30:24.272735	3.52
13	16:30:25.530374	3.3
14	16:30:38.388819	3.52
15	16:30:39.273206	3.59



(a)



(b)

Gambar 4.32 (a) Hasil 3D *analyzer* di *google earth* dan (b) hasil pemetaan kadar gas pengujian ketiga.

Analisis :

Quadcopter mampu mengikuti sesuai dengan titik-titik *waypoint* yang dideklarasikan. Namun terdapat error jarak GPS pada *waypoint* 3 sebesar 66 cm dari pembacaan GPS di *mission planner* dengan kenyataan. Error ketinggian berdasarkan *set point* awal sebesar 15,75%. Data peta dari 3D *analyzer* sesuai dengan hasil jalur yang dilewati oleh *quadcopter* sesuai gambar 4.32a dan 4.32b. Terdapat perbedaan kadar gas saat di permukaan tanah dengan di udara ± 3 ppm. Hal ini dikarenakan interferensi hembusan angin di sekitar tempat pengujian.

4.4.4 Pengujian Sistem Ke-4

Pengujian kedua dilakukan pada sebuah lapangan di tempat parkir Teknik Industri dengan ukuran 30x15 meter. Gas polutan berasal dari hasil pembakaran sampah dan polutan knalpot 2 sepeda motor pada titik tertentu seperti pada gambar 4.33.



Gambar 4.33 Lokasi pengujian dan posisi *quadcopter* saat awal misi keempat

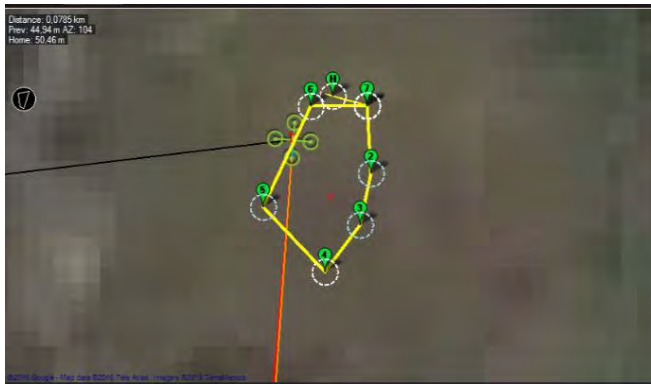
Pada pengujian berikut, nilai ketinggian (*altitude*) ditetapkan setinggi 3 meter. Sampah dibakar dan sepeda motor dinyalakan untuk menghasilkan gas polutan motor. Hasil pembacaan CO-meter di sekitar tempat pembakaran berkisar antara 2-6 ppm. Data kadar gas yang terukur (gambar 4.34) dibandingkan dengan hasil pengukuran oleh *quadcopter*.



(a)

(b)

Gambar 4.34 (a) Hasil pembacaan CO-meter di polutan pembakaran sampah dan (b) hasil pembacaan di sepeda motor.



Gambar 4.35 Deklarasi *waypoint* ketiga dan tampilan *google maps* di *mission planner*

Pada tabel 4.17, didapatkan nilai dari *longitude* (busur), *latitude* (lintang), dan *altitude* saat *quadcopter* melakukan misi *waypoint*. Dari hasil pengujian, *quadcopter* berhasil mengikuti semua titik *waypoint*

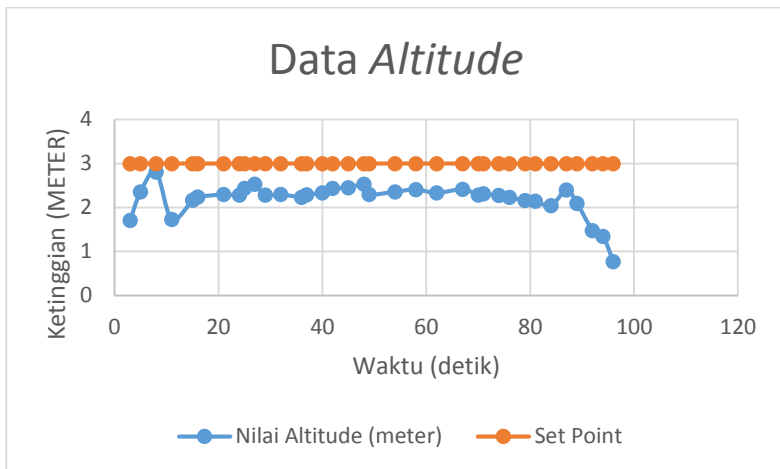
Tabel 4.17 Nilai GPS Pengujian *Waypoint*

No	Jenis Misi	Altitude	Latitude	Longitude
1	Waypoint 1	4	-7,2843157	112,7974878
2	Loiter (WP 2)	4	-7,2844088	112,7974932
3	Loiter (WP 3)	4	-7,2844793	112,7974597
4	Waypoint 4	4	-7,2845312	112,7973993
5	Loiter (WP 5)	4	-7,284455	112,797342
6	Waypoint 6	4	-7,2843157	112,7974087
7	Land	0	-7,2843144	112,7974878

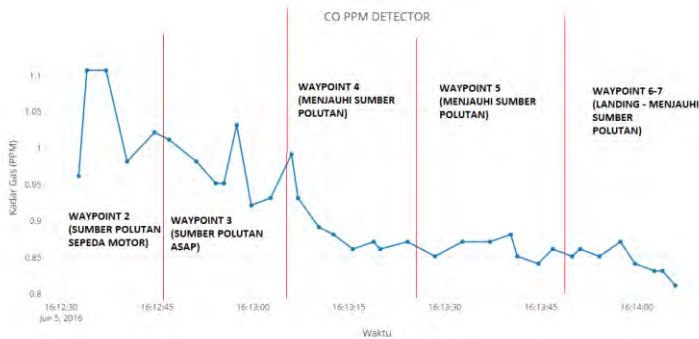
Tabel 4.18 Nilai sensor barometer saat pengujian keempat

Waktu(Detik)	Nilai Altitude (meter)	Set Point	Error (%)
15	2,16679	3	27,77
16	2,242084	3	25,26
25	2,434502	3	18,84
27	2,529316	3	15,68
36	2,233718	3	25,54
37	2,28914	3	23,69
40	2,331321	3	22,28
48	2,529316	3	15,68
58	2,409404	3	19,68
67	2,412192	3	19,59
Error Rata- Rata			21,4

Data sensor barometer menunjukkan hasil data error ketinggian seperti pada tabel 4.18 dan gambar 4.36. *Set point* ketinggian pada 3 meter, namun hasil di kenyataan terdapat error sebesar 21,4% saat misi *waypoint*. Gambar 4.37 menunjukkan hasil data logger pada *plotly*.



Gambar 4.36 Error ketinggian *quadcopter* berdasarkan sensor Barometer



Gambar 4.37 Hasil Data Logger Pengujian Keempat di *Plotly*

Tabel 4.19 Data kadar gas PPM

No	Waktu	Kadar Gas
1	2016-06-05 16:12:32.761815	0.96
2	2016-06-05 16:12:34.033867	1.105
3	2016-06-05 16:12:37.046376	1.105
4	2016-06-05 16:12:51.171120	0.98
5	2016-06-05 16:12:54.183637	0.95
6	2016-06-05 16:12:55.455851	0.95
7	2016-06-05 16:13:10.314474	0.89
8	2016-06-05 16:13:12.592885	0.88
9	2016-06-05 16:13:15.604137	0.86
10	2016-06-05 16:13:32.741573	0.87
11	2016-06-05 16:13:37.026549	0.87
12	2016-06-05 16:14:04.004632	0.83
13	2016-06-05 16:14:06.011091	0.81

Agar mendapatkan hasil yang diharapkan, sistem *quadcopter* diuji saat berada di posisi *ground* seperti gambar 4.38a dan 4.38b. Dimana

quadcopter dan CO-meter diletakkan di dekat sumber polutan. Sehingga diperoleh data perbandingan selisih pengukuran antara sensor gas MQ-7 dengan CO-meter. Tabel 4.20 menunjukkan error selisih rata-rata dari 10 sampel sebesar 1,11 ppm.



(a)



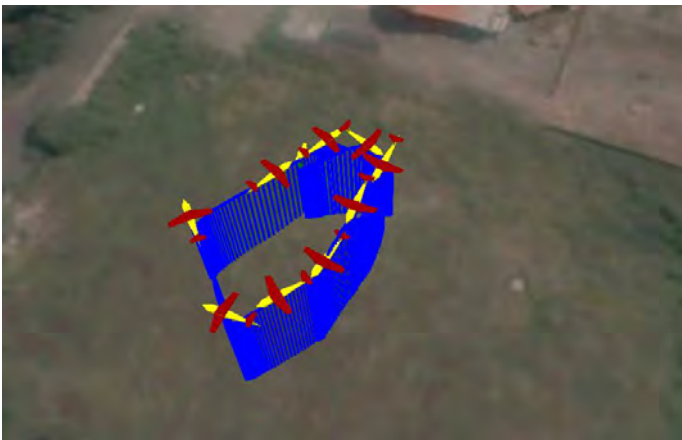
(b)

Gambar 4.38 Pengujian saat *quadcopter* di *ground*, (a) CO-meter yang terukur, (b) *quadcopter* saat *throttle* awal dan menyedot gas

Tabel 4.20 Data kadar gas PPM saat di *ground*

No	CO-meter (ppm)	Sensor Gas (ppm)	Selisih ppm
1	0	0,6	0,6
2	0	0,8	0,8
3	0	1,3	1,3
4	1	1,5	0,5
5	4	1,4	2,6
6	4	1,7	2,3
7	2	1,3	0,7
8	2	1,4	0,6
9	0	1,4	1,4
10	0	1,3	1,3
Error Selisih Rata-Rata			1,11

Dapat disimpulkan bahwa terdapat selisih error pembacaan sensor gas MQ-7 dengan CO-meter di *ground* sebesar $\pm 0,5-3$ ppm. Faktor-faktor lain yang menyebabkan error pembacaan gas adalah interferensi angin dari tempat sekitar pengujian. Selain angin, karakteristik CO-meter yang memiliki spesifikasi seperti sensor TGS. Dimana CO-meter susah untuk mendeteksi gas CO apabila tidak dilengkapi kipas penyedot.



(a)



(b)

Gambar 4.39 (a) Hasil 3D *analyzer* di *google earth* dan (b) hasil pemetaan kadar gas pengujian keempat.

Analisis :

Quadcopter mampu mengikuti sesuai dengan titik-titik *waypoint* yang dideklarasikan. Terdapat error ketinggian sebesar 2,7% dari *set point* altitude. Data peta dari 3D *analyzer* sesuai dengan hasil jalur yang dilewati oleh *quadcopter* sesuai gambar 4.39a dan 4.39b. Terdapat perbedaan kadar gas saat di permukaan tanah dengan di udara ± 2 ppm. Hal ini dikarenakan interferensi hembusan angin di sekitar tempat pengujian dan posisi *quadcopter* berada di atas sumber asap.

4.5 Evaluasi Sistem

Pengujian sistem secara keseluruhan *quadcopter* bergerak secara otomatis berdasarkan titik-titik *waypoint* yang dideklarasikan. Quadcopter dapat menempuh jarak 25 meter dalam waktu 17 detik, sehingga kecepatan rata-rata quadcopter 1,47 m/s. Data logger berupa web server plotly mampu menampilkan data polutan gas berupa PPM secara *online real time*. Kontrol PID menggunakan nilai P,I,D standar dari mission planner.

Pendeteksian gas menggunakan sensor gas saat ketinggian 3 meter terdapat perbedaan saat pengukuran di permukaan tanah sebesar

$\pm 2-4$ ppm. Saat di darat, perbandingan selisih antara CO-meter dengan sensor gas di *quadcopter* adalah $\pm 0-2$ ppm. Terdapat error ketinggian *quadcopter* dibandingkan dengan *set point altitude*. Dimana error ketinggian pengujian ketiga sebesar 15,75% dan pengujian keempat sebesar 21,4%. Sehingga error rata-rata ketinggian sebesar 18,575%. Hal ini menyebabkan saat melakukan misi *waypoint*, *quadcopter* terkadang naik-turun dengan sendirinya. GPS memiliki nilai error posisi 20-66cm saat di lapangan luas. Pada tempat yang terdapat gedung, error posisi GPS bisa mencapai 10 meter.

BAB V

PENUTUP

5.1 Kesimpulan

Quadcopter merupakan salah satu jenis UAV (*Unmanned Aerial Vehicle*) jenis rotary. Sensor gas semikonduktor mampu mendeteksi kadar gas polutan yang ada di udara. Dari perancangan, realisasi, dan pengujian alat pada tugas akhir ini dapat disimpulkan bahwa :

1. *Quadcopter* dapat mengangkat beban seberat 2,5 kg, namun dengan nilai *throttle* awal 65% *quadcopter* mampu melayang.
2. Metode regresi linier dengan fungsi *power* dapat digunakan untuk mencari model matematis dapat diterapkan untuk kalibrasi sensor gas ke satuan *part per million* (ppm) dengan error 7,25 %.
3. Sensor gas MQ-7 mendeteksi gas berupa polutan karbon monoksida (CO) dengan error rata-rata 9,4 % dibandingkan CO-meter. Error terjadi karena kurangnya asap polutan, sehingga sensor gas MQ-7 mendeteksi sedikit nilai CO di udara.
4. Besar-kecilnya polutan mempengaruhi pengukuran gas CO di *quadcopter*. Terdapat selisih kadar gas sebesar 2-3 ppm saat *quadcopter* berada di dekat polutan dengan di udara. Namun saat *quadcopter* di darat dan diletakkan dekat polutan, mendapatkan selisih kadar gas $\pm 0,5-3$ ppm dibanding CO-meter. Peletakan sensor gas di *quadcopter* mempengaruhi pembacaan sensor gas.
5. Dari hasil pengujian, didapatkan hasil error *waypoint GPS* sebesar 20-66 cm pada daerah lapang tanpa bangunan, pada daerah yang dikelilingi bangunan bisa mencapai error 10 meter. Data error rata-rata ketinggian sebesar 18,575%.
6. *Plotly* berupa web server raspberry pi 2 mampu menampilkan data kadar gas ppm secara *online real time*. *Data logger plotly* mampu menyimpan data di *web server* hingga 500 data saja. Hal ini karena fitur *free user* yang dibatasi.

5.2 Saran

Dalam mendesain sebuah rangka mekanik *quadcopter* diperlukan bahan yang kokoh dan ringan, karena kemampuan rangka *quadcopter* yang baik dapat memberikan gerakan dan gaya dorong ke bawah (*thrust*) *quadcopter*. Pengujian dilakukan dengan memberi gas polutan yang

banyak sehingga sensor gas MQ-7 mampu mendeteksi gas CO pada ketinggian tertentu. Untuk keberlanjutan sistem monitoring *real time*, data posisi gps dan data kadar gas (ppm) diharapkan dapat ditampilkan dalam satu *software*. Sehingga mempermudah pengguna saat monitoring *quadcopter*.

DAFTAR PUSTAKA

- [1]. Purwitaningsih, Santika, “Polusi Udara dan Kaitannya dengan Ekonomi Kota Surabaya”, Surabaya, 2015.
- [2]. Neumann, Patrick P, “Gas Source Localization and Gas Distribution Mapping with a Micro-Drone”, BAM Bundesanstalt für Materialforschung und –prüfung, Berlin, 2013.
- [3]. Panjaitan Berkat, Siringo-Ringo Berliana, dkk, “Sensor Gas”, Medan: Technology Science, Medan, 2013.
- [4]. Sparkfun, “Technical Data MQ-7 Gas Sensor” <URL: <https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-7.pdf>>, 2012.
- [5]. Arya Adi Saputra dan Andi Dharmawan, “Rancang Bangun Quadcopter untuk Pemantauan Kadar Karbon Monoksida di Udara”, UGM, Yogyakarta, 2013.
- [6]. Jati Utomo, Bimo, “Rancang Bangun UAV (Unmanned Aerial Vehicle) Model Quadcopter Dengan Menggunakan Algoritma Proportional Integral Derivative”, Universitas Telkom, Bandung, 2014.
- [7]. Rengrajan, M., “Algorithm Development and Testing of Lowcost Waypoint Navigation System”, Proceeding of IRACST, Pune, India, 2013.
- [8]. 3Drobotics, “Pixhawk Manual rev 7”, <URL: <http://3drobotics.com/wp-content/uploads/2014/03/pixhawk-manual-rev7.pdf>>, 2014.
- [9]. Guadulape Crespo dan Javier Garriado, “Setup of a Communication and Control System of a Quadrotor type Unmanned Aerial Vehicle”, Univ. Autonoma de Madrid, Spain , 2014.
- [10]. ST-Electronics, ”STM32F4 Discovery” <URL:<http://www.st.com/web/catalog/tools/FM116/SC959/SS1532/PF252252>>, 2015.

BIODATA PENULIS



Irfan Fachrudin Priyanta dilahirkan di Bantul, pada tanggal 30 Agustus 1994 dari pasangan Bapak Dwi Priyanta dan Ibu Koesmiyati. Penulis adalah anak pertama dari 2 bersaudara. Penulis menyelesaikan pendidikan dasar di SDN Pucang II Sidoarjo, dilanjutkan dengan pendidikan menengah di SMPN 19 Surabaya dan SMAN 5 Surabaya. Pada tahun 2012, penulis memulai pendidikan di jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS) Surabaya. Selama kuliah, penulis aktif dalam Unit Kegiatan Mahasiswa (UKM) Robotika sebagai Ketua UKM

periode 2014-2015. Penulis turut andil sebagai asisten Laboratorium Elektronika Dasar dan menjadi koordinator asisten laboratorium pada periode 2015-2016.

Email : irfanfachrudin7@gmail.com, irfan.fachrudin12@mhs.ee.its.ac.id